

Topological Data Analysis in Image Processing

Jakob Rode, Jayden West, Marko Marceta

June 2025

Abstract

In this paper, we will summarize some recent advancements in the intersection of topological data analysis and image processing and then proceed to our own case study. We will establish the necessary mathematical background, examine various medical and scientific applications, and discuss the differences and similarities between TDA and more traditional machine learning methods. We have compiled a dataset of cartoon characters on which we will demonstrate some of the methods discussed in the paper using python and the gudhi library.

1 Introduction

Over the last few decades, the field of topological data analysis has emerged in order to understand increasingly complex and high-dimensional data. TDA provides tools to extract meaningful geometric and topological features, such as connected components, loops, and holes from data, without relying on predefined coordinates or metrics. Put simply, it allows one to study the "shape" of data.

While topology, and in particular algebraic topology, the mathematical background of TDA has been around for over a century, applying these rigorous methods to real data is still a fairly new concept. Some of the early pioneers of the field include Herbert Edelsbrunner and Gunnar Carlsson in the early 2000's with the formulation of the persistence diagram, and it's intuitive visualizations as persistence diagrams and barcodes. Today, although still in its infancy, TDA is accepted as a useful tool in the scientific community for analyzing complex high dimensional point clouds. Modern imaging technologies generate vast amounts of high-dimensional, often noisy data that challenge conventional analysis techniques. From medical scans and microscopy to satellite and materials imaging, the task of extracting robust, meaningful features from complex image structures is a task well suited for topological data analysis.

2 Mathematical Background

The mathematics underlying topological data analysis are advanced and mostly out of the scope of this paper, however we will define a few of the most important concepts.

Let V be a finite set of vertices. A k -simplex is denoted as the set

$$\sigma = \{v_0, v_1, \dots, v_k\} \subseteq V$$

where each subset of σ is also a face of the simplex. A simplicial complex K over V is a collection of simplices such that:

1. If $\sigma \in K$, then every face $\tau \subseteq \sigma$ is also in K ,
2. Every vertex of $\sigma \in K$ belongs to V .

A widely used construction is the Vietoris–Rips complex. Given a finite point cloud $X \subseteq \mathbb{R}^n$ and a scale parameter $\epsilon > 0$, the Vietoris–Rips complex $\text{VR}_\epsilon(X)$ contains a k -simplex for each subset of $k + 1$ points whose pairwise distances are all $\leq \epsilon$. A filtration is a nested sequence of simplicial complexes built from data, where each complex K_ϵ includes more simplices as the scale parameter ϵ increases.

Let $\{K_\epsilon\}_{\epsilon \geq 0}$ be a filtration of simplicial complexes:

$$K_0 \subseteq K_{\epsilon_1} \subseteq K_{\epsilon_2} \subseteq \dots \subseteq K_{\epsilon_n}$$

For each scale ϵ_i , we construct a simplicial complex K_{ϵ_i} from the data and examine its topological features—such as the number of connected components, loops, or enclosed voids. As ϵ increases, more connections form between data points, causing new features to appear and existing ones to disappear.

Persistent homology tracks these features across the entire sequence of scales. If a feature appears at scale ϵ_i and disappears at scale ϵ_j , we record this with a persistence interval:

$$[b, d) \quad \text{where } b = \epsilon_i, \quad d = \epsilon_j$$

These intervals are collected into a persistence barcode or a persistence diagram, which visually summarize the lifespan of topological features across scales. Features that persist over a large range of ϵ are typically considered meaningful, while short-lived features are often treated as noise.

This is far from a complete study of the mathematics behind topological data analysis, but it does lay some of the groundwork to contextualize the content of this paper.

3 From Images to Point Clouds

In this paper, we want to apply TDA techniques to the processing of images. In order to do this however, we need to encode our image as a point cloud, which in its own is not a trivial task. A picture taken with a digital camera is generally stored as a two dimensional array where the values are either RGB values (color images) or pixel intensities (grayscale images).

The most natural way to embed a grayscale image is to consider $\{(x, y, I)\} \subset \mathbb{R}^3$ where x, y are the array indices and I is the grayscale intensity. Similarly for an RGB image we may encode it as $\{(x, y, R, G, B)\} \subset \mathbb{R}^5$

Additionally we may discretize our intensity values to binary values, down sample by picking every n -th pixel or using a uniform random sample, or compute graph based embeddings of the image.

If we are working with a whole collection of images however, we may want to consider each image as a single point in the high dimensional space \mathbb{R}^{m^2} for an image of $m * m$ image. For example, a $16 * 16$ grayscale image becomes a vector in \mathbb{R}^{256} by flattening the 2D array into a 1D vector, where each array element corresponds to a pixel intensity.

These are just a few methods for encoding our images as point clouds in order to process them for topological data analysis. In practice, the method once chooses is highly contextual and depends on the particular application or problem one is trying to solve.

4 Satellite Cloud Imaging Case Study

In this section, we will summarize the methods used by Ver Hoef L, Adams H, King EJ and Ebert-Uphoff to classify meoscale clouds. The ability to efficiently classify cloud types is an important problem in environmental sciences that can be directly applied to weather prediction and climate monitoring models. Using traditional methods, this can be challenging due to a lack of well-labeled data sets. In this study, the authors aim to use TDA techniques on a dataset of 100,000 images to classify four different cloud meoscale cloud types, sugar, flower, fish or gravel. The images are originally in RGB form, but the authors aim to simplify the dataset by converting to grayscale using the formula

$$I = 0.299R + 0.587G + 0.144B$$

which is a common method for measuring perceived brightness of an RGB image as grayscale. In order to further reduce computational complexity, the authors randomly sample six 96×96 pixel subsets from each annotated image. For every subsample, the authors compute the superlevelset persistent homology in 0 and 1 dimensions, which are then vectorized as a point in \mathbb{R}^{2000} . Then the six points for an annotated image are averaged in order to obtain a single embedding for each annotation. These vectors then serve as the inputs for a support vector machine based classifier. By projecting the SVM's decision boundaries back into the topological space, the study identifies which topological features distinguish different cloud patterns. For instance, sugar clouds exhibit numerous small, bright components against a dark background, leading to prominent 0-dimensional features.

The results are quite impressive. The TDA-based method effectively differentiates between certain cloud types (e.g., sugar vs. flowers) but struggles with others (e.g., fish vs. flowers), likely due to the scale of distinguishing features relative to the subsample size.

This approach is computationally efficient as we are just taking small samples of the large dataset, requires significantly fewer labeled samples compared to deep learning methods and the method's deterministic nature allows for predictable behavior and understanding of failure modes. The study clearly shows the power of persistent homology in extracting textural differences in satellite imaging.

5 Another Example/Summary

6 Cartoon Characters and TDA

In this section, we now move on to our own example where we apply a few basic TDA methods to a dataset we compiled of about 200 images of cartoon characters. In the analysis of each character, we compute our persistence diagram by creating a grayscale image using a sublevel set filtration. We then apply this to a cubical complex. Each pixel in the image is treated as a single cell, with a grayscale value determining its intensity. The levels of intensity provide the values for the filtration. Using this type of sublevel filtration includes lower-intensity regions of the image, starting with the darkest pixels and gradually moving to the brighter pixels.

As the filtration value increases, new topological features are created and killed—connected components such as holes appearing (“birth”) and eventually merging (“death”). The persistence diagram provides a visual summary of these features, plotting each one as a point where the x-coordinate represents the birth and the y-coordinate represents its death. The features that live longer are plotted farther from the diagonal. These points are typically more significant and meaningful within the image, whereas the short-lived features closer to the diagonal often represent just random noise.

This approach enables us to capture and quantify certain information in the data that may be difficult to detect with other traditional methods. The persistence diagram also allows us to assign weighted values to topological features based on their lifespan. Transforming these values into numerical features enables machine learning models to learn from these summary statistics.

6.1 The topology of Homer Simpson

