

Topology of a Smile: Persistent Homology in Dental Imaging

Leon Dahlmeier

DLEON@ETHZ.CH

Department of Mathematics, ETH Zürich, Zürich, Switzerland

Sara Kališnik

SKALISNIK@PSU.EDU

Department of Mathematics, PSU, Pennsylvania, United States

Albert Mehl

ALBERT.MEHL@ZZM.UZH.CH

Department of Computerized Dentistry, Clinic of Conservative and Preventive Dentistry, University of Zürich, Zürich, Switzerland

Bastian Rieck

BASTIAN.GROSSENBACHER-RIECK@UNIFR.CH

Department of Computer Science, University of Fribourg, Fribourg, Switzerland

Abstract

CBCT (Cone Beam Computed Tomography) scans provide detailed three-dimensional images, widely used in dentistry for diagnostic and treatment planning tasks. While invaluable, analyzing and documenting these scans is labor-intensive, prompting efforts to automate key steps like the classification and segmentation of anatomical structures to identify tooth types and associated pathologies. In this article, we propose an approach to automation that leverages persistent homology, a framework from topological data analysis that studies the shape of data by identifying features like connected components, holes, and voids across multiple scales. Persistent homology, together with a support vector machine, allows us to classify teeth in a CBCT scan and to perform diagnostics. Our method advances the state of the art, reaching average accuracy scores of 97.67% for tooth-labeling and 96.77% for diagnostic tasks, outperforming a CNN trained on the same data with accuracy of 70.27% and 86.67%, respectively.

Keywords: Dental Cone Beam Computed Tomography, Topological Data Analysis, Machine Learning, Tooth Identification, Dental Diagnostics.

1. Introduction

Three-dimensional X-ray images (Cone Beam Computed Tomography, CBCT) are used, for example, before implant and orthodontic treatments, prior to the extraction of impacted and fractured teeth, and for classifying cystic lesions and other pathological findings, among other applications (Lee et al., 2020; Morgan et al., 2023). Additionally, the documentation can be used to monitor changes over time, thereby predicting the progression of diseases or controlling treatments (Morgan et al., 2023). CBCT data incorporate a wealth of information; however, for dentists, the 3D visualization is laborious and time-consuming; hence, there are various approaches to automatically extract relevant findings and generate documentation. Key steps in this workflow are the segmentation and the classification of anatomical structures (Fontenele et al., 2023; Xiang et al., 2024). In particular, it is crucial to identify the types of teeth present in the current CBCT and determine if they exhibit any specific alterations or pathological findings (Xiang et al., 2024). Owing to the rise of deep neural networks, approaches for extracting information and pathological findings have been tested and implemented for a variety of radiographic images. In dentistry, the primary focus

has traditionally been on 2D radiographs, such as panoramic images and bitewings (Hamamci et al., 2023). However, in recent years, 3D X-ray imaging has gained increasing attention. Attempts to solve the automatic labeling task in CBCT data were described by Mikia et al. (2017) and Gerhardt et al. (2022). More general approaches combine tooth *labeling* with tooth *segmentation* (Cui et al., 2019; Ezhov et al., 2019; Shaheen et al., 2021; Zhou et al., 2025). A typical setting is described by Cui et al. (2022), which performs tooth classification and segmentation of teeth and bony anatomical structures, reaching an accuracy (sensitivity) of up to 93%. The common denominator of all these works is their use of convolutional architectures, making use of CNNs or more complex models like UNet (Polizzi et al., 2023). This makes the approaches infeasible to train or apply in data-scarce scenarios.

In this article, we introduce an automatic classification procedure using tools from topology, a branch of mathematics that studies the shape of objects. In particular, we employ persistent homology, an invariant that captures topological features across scales and has been shown to be effective for image-based analysis and classification (Adcock et al., 2014; Carlsson, 2014; Edelsbrunner et al., 2002; Levenson et al., 2024; Vandaele et al., 2023). We combine these topological descriptors with support vector machines to create a novel method for classifying teeth and pathological findings in CBCT data. Unlike existing techniques, our method (i) requires far fewer features, while being able to handle regions of interest (ROIs) of *varying size*, (ii) is provably stable (Cohen-Steiner et al., 2007; Turkeš et al., 2021), and (iii) captures morphological features (Peng et al., 2024; Singh et al., 2023).

The results of our method are compared with an existing CNN-based approach (Esmailyfard et al., 2024), achieving accuracy scores of 97.67% for tooth labeling and 96.77% for diagnostic tasks, thereby outperforming a CNN trained on the same data, which attains accuracies of 70.27% and 86.67%, respectively.

2. Methods

2.1. Topological Persistence

This section provides a brief and informal introduction to cubical complexes and persistent homology. For an in-depth treatment of the subject, see Carlsson (2014) or Edelsbrunner et al. (2002).

2.1.1. CUBICAL COMPLEXES

Our data, which consists of CBCT scans (as seen in Figure 5), is given in the form of a stack of grayscale images. To represent such images, we use *cubical complexes*. Informally, an m -dimensional cubical complex consists of k -dimensional elementary cubes (where $k \leq m$) glued together along common boundaries; 0-dimensional, 1-dimensional and 2-dimensional elementary cubes are vertices, edges and squares, respectively. For the formal definition, please refer to Kaczynski et al. (2004).

From an n -dimensional image, we construct a cubical complex by assigning an n -dimensional elementary cube to each n -pixel. We then include all lower-dimensional boundary cubes.

To incorporate grayscale information, we consider a one-parameter family of cubical complexes rather than a single complex, thereby preserving information that would otherwise be lost in a single-complex representation.

2.1.2. CUBICAL FILTRATIONS

A one-parameter family of cubical complexes together with inclusion maps is called a (*cubical*) *filtration* (Carlsson, 2014). Given an n -dimensional image and a function f defined on the pixels (e.g., grayscale values), we construct a cubical complex C_t by including an n -dimensional cube for each pixel p with $f(p) \leq t$, together with all boundary cubes. As t increases, additional pixels are included until C_t contains a cube for each pixel. This yields a nested sequence

$$\emptyset = C_0 \subseteq C_1 \subseteq C_2 \subseteq \cdots \subseteq C_n = C, \quad (1)$$

referred to as a *sublevel set filtration*. In this paper, we consider the following images and filter functions for constructing cubical filtrations: the *original image*, the *inverted image*, the *contrast image*, and the *inverted contrast image* (see the images on the left in Figure 1). The inverted image is generated by setting each pixel’s grayscale value g_o to $g = g_{\max} - g_o$, where $g_{\max}$ denotes the highest grayscale value in the original image. The contrast image is generated by setting each pixel’s old grayscale value g_o to $h = |g_o - 0.5 \cdot (g_{\min} + g_{\max})|$. For the inverted contrast image, we first take the contrast image of the original image, and then the inverted image of the contrast image. The choice of contrast image is inspired based on considering carious lesions, as they appear darker than the neighboring healthy enamel or dentin but may still be brighter than surrounding soft tissue or air, making it harder for the lesion to appear as a whole in the sublevel set filtrations of the original scan.

To illustrate this cubical filtration procedure, consider Figure 1. The top row shows the grayscale sublevel set filtration of the original image, while the remaining rows correspond to the inverted, contrast, and inverted contrast images. As the parameter increases in the top row, additional cubes are added according to pixel intensity, revealing topological features such as loops. The evolution of these features across the filtration is captured using persistent homology.

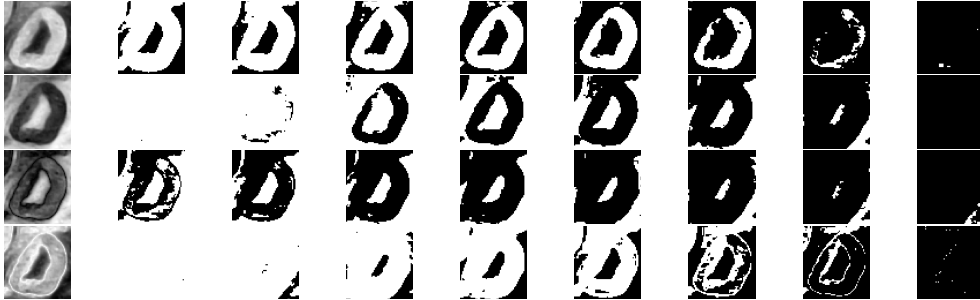


Figure 1: Sublevel filtration of the original, the inverted, the contrast, and the inverted contrast image, respectively.

2.1.3. PERSISTENT HOMOLOGY

Homology is a topological invariant used to distinguish spaces by considering patterns within them. For a cubical complex X , the k -dimensional homology group $\mathbf{H}_k(X)$ encodes information about k -dimensional voids, and its dimension, the k -th Betti number $\beta_k(X)$, counts these features. In particular, $\beta_0(X)$ counts connected components, $\beta_1(X)$ counts loops, and $\beta_2(X)$ counts two-dimensional voids.

Persistent homology is an adaptation of homology to the setting of filtrations. It allows us to track when topological features, such as connected components, loops and voids, appear and disappear in a filtration (Carlsson, 2014; Edelsbrunner et al., 2002). A *persistence diagram* associated with a filtration is a collection of points $\{(b_l, d_l)\}_{l \in L}$ in the extended plane $\overline{\mathbb{R}}^2$, where $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$. These points lie on or above the diagonal, with birth times b_l as the x -coordinates and death times d_l as the y -coordinates. The multiplicities of points on the diagonal are set to ∞ . The lifetime of the l -th feature is given by $d_l - b_l$. Figures 3 and 4 depict persistence diagrams arising from sublevel set filtrations for both the original image in Figure 2 and its inverted counterpart. In these diagrams, red points represent dimension-zero homology, while blue points indicate dimension-one homology.

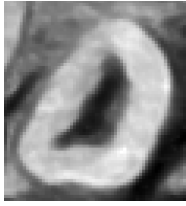


Figure 2: Image of a two-dimensional slice of a tooth, specifically 17.

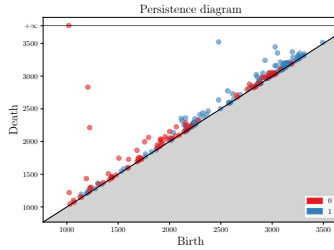


Figure 3: The persistence diagram for the sublevel set grayscale filtration of the image in Figure 2.

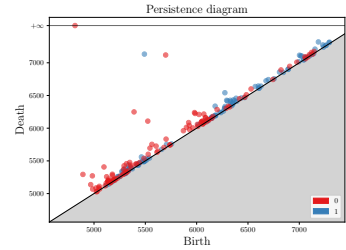


Figure 4: The persistence diagram for the sublevel set filtration of the inverted image of Figure 2.

2.1.4. VECTORIZATION TECHNIQUES

While persistent homology is a powerful tool for extracting topological features from data, its output is non-Euclidean, making it difficult to integrate with standard machine learning methods that require vector inputs. Various vectorization techniques (Adams et al., 2017; Bubenik, 2015; Kališnik, 2018) address this by mapping persistence diagrams to fixed-dimensional representations while preserving topological information. For our dataset, *persistent statistics* (Ali et al., 2023) performed best, generating feature vectors from summary statistics of birth and death times (e.g., mean, standard deviation, median, and percentiles). Appendix A.4 compares this approach with several alternative vectorization methods and supports our choice.



Figure 5: A CBCT scan represented by an axial, coronal and sagittal slice.

2.2. Data and Study Design

2.2.1. DATA

Twenty CBCT datasets, we have depicted one in Figure 5, were randomly extracted from a larger data base of fully anonymized CBCT cases ($n = 250$). These cases were sourced from the X-ray archiving system of a university clinic, which includes patient cases used for examinations or surgical treatment planning as part of routine clinical practice (anonymized Ethics Committee). The inclusion criterion for the 20 CBCT datasets was the presence of a nearly full dentition (≥ 24 teeth (excluding wisdom teeth)) and a dentation in occlusion. Each tooth was labeled with its FDI-label (Nelson, 2019) and associated diagnoses. This was performed by a dentist with more than three years of clinical experience. Our dataset contains diagnoses as follows. There are 51 fillings: 7 mesial, 42 occlusal, and 2 distal. Additionally, 44 teeth are held in place by a retainer. Among the 26 impacted teeth, 6 are covered only by gingiva, while 20 are also covered by bone. There are 7 cases of secondary caries and 3 cases of attrition. Finally, 476 teeth show no clinically relevant findings. In total, we use 600 teeth to perform diagnostics. For the FDI-labeling task, we exclude wisdom teeth, as too many of them are impacted. All other teeth are in occlusion, resulting in 555 teeth available for the FDI-labeling task.

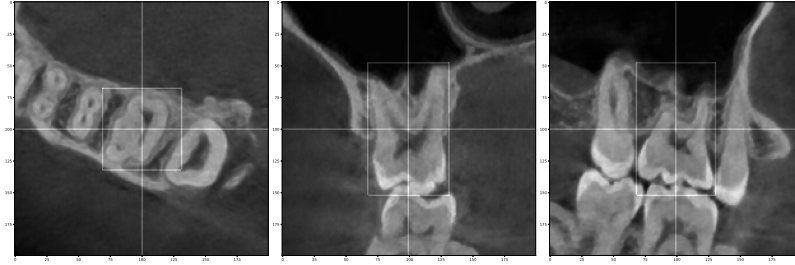


Figure 6: Three two dimensional slices through a CBCT-scan, tooth 16 is depicted within its ROI and surrounding structures are included.

2.2.2. STUDY DESIGN

Our proposed *pipeline* to classify the CBCT data is as follows:

1. **Generating ROI from CBCT-data** We generated regions of interest (ROIs) containing individual teeth, as seen in Figure 6, via coarse segmentation of dental structures (teeth, maxilla, mandible) using a deep learning approach. For this step, a 3D U-Net, implemented in MONAI (v1.3) with PyTorch (v2.1), was trained on 120 CBCT datasets with 15 for validation. Standard preprocessing and 3D data augmentation were applied. The model was trained with the Adam optimizer and a Dice–Cross-Entropy loss. After training, the network was used to segment new CBCT volumes (NIfTI) (Swiss National Foundation, 2022). From each coarse tooth segmentation, a 3D bounding box was computed, expanded by 3 voxels in all directions to ensure full coverage, and the corresponding sub-volumes (ROIs) were extracted from the CBCT data. For more information, see Appendix A.2.
2. **Constructing Filtrations** For classification, we use the full three-dimensional ROI together with forty-one two-dimensional slices in the coronal, sagittal, and axial directions. For details, see Appendix A.3. We do not perform any normalization of the grayscale values, as this outperforms all normalization attempts (see Appendix A.1). From the three-dimensional ROI and the slices, we consider the **original**, **inverted**, **contrast**, and **inverted contrast images** and construct the corresponding sublevel set filtrations as described in Subsection 2.1.2.
3. **Computing Persistence Diagrams** From the filtrations obtained in Step 2 we construct persistence diagrams in each dimension less or equal to the one of the image or volume respectively, resulting in $(4 \cdot 3) + ((41 \cdot 3) \cdot 4 \cdot 2) = 996$ persistence diagrams in total per ROI.
4. **Vectorization** To assign vectors to persistence diagrams, we use the code provided in (Ali et al., 2023), specifically **persistent statistics**. Doing so, we extract 38 features out of each persistence diagram (mean or quantiles of births or deaths, etc.). For the results obtained with the other vectorization techniques, see Appendix A.4.
5. **SVM** Through persistent statistics we generate 37848 features. In order to reduce the number of features, we train a linear SVM with $C = 0.001$ and order the features by their assigned weights. We then only use the features with the highest weights to perform 5-fold cross-validation, iteratively increasing the number of used features. We keep track of the highest accuracy; once the score does not improve by adding more features, we stop. Appendix A.5 summarizes additional experiments, including comparisons of SVM kernels, the incorporation of texture-analysis features, and several manual feature-selection strategies, none of which improved the test accuracy.
6. **Iterated labeling** In order to decrease the number of labels that need to be differentiated at once, we perform iterative steps, each narrowing down the final label. The individual steps are described in Figure 7.

To perform the FDI-labeling, we first differentiate between the upper and lower jaws, then identify the quadrant, followed by determining whether the tooth is an incisor, canine, premolar or molar, and finally assign the appropriate label. This can be seen in Figure 7(a). Note that this allows us to first have binary decision then a quaternary decision and finally

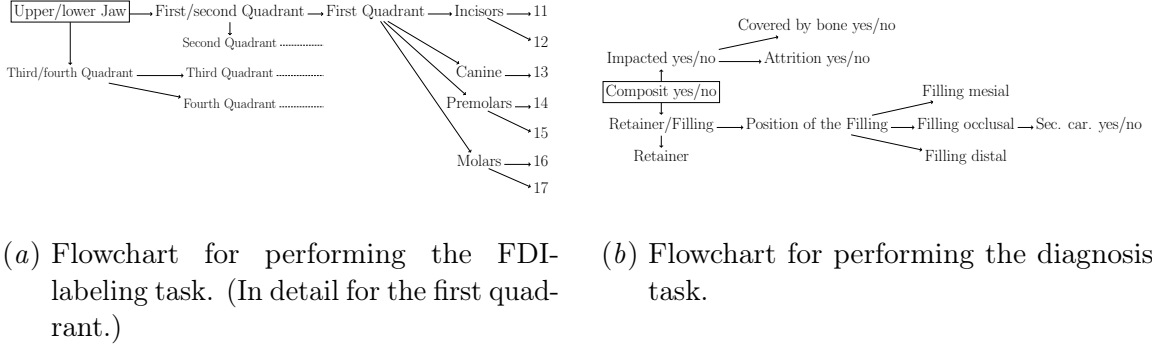


Figure 7: Flowcharts for performing the FDI-labeling and diagnosis tasks.

again a binary, or none at all if we have a canine. Additionally, this allows us to mirror the teeth of the second and third quadrant along the sagittal plane for the final two steps, in order to double the amount of samples we have in each quadrant.

To perform a diagnosis, we first check whether composite material is present in the region of interest. If not, we determine whether the tooth is impacted. If it is, we differentiate between teeth covered by bone and those covered only by gingiva. Otherwise, we assess for attrition damage. If composite material is present in the region of interest, we begin by distinguishing between retainers and fillings. For fillings, we also evaluate their position and check whether the tooth is affected by secondary caries. This can be seen in Figure 7(b).

In both tasks, we initially start with the whole dataset, meaning $n = 555$ for the FDI-labeling task and $n = 600$ in order to perform diagnostics. Each step reduces the dataset, since teeth excluded at earlier stages are removed from subsequent classification steps. For example, in the decision step whether the given tooth is in the first or second quadrant we only use teeth from the upper jaw, as the previous step already differentiated between upper and lower jaw. The only exception is the step in which we differentiate between incisor, canine, premolar and molar, as we mirrored the teeth from the other quadrant in order to double our sample size. In each step, we use 5-fold cross-validation to split the data in training and testing.¹

3. Results

The scores and number of features used for each individual decision step can be seen in Tables 1 and 2. The resulting scores for each label can be seen in Table 3. In the FDI-labeling task, this results in scores ranging from 96.60% to 98.57% averaging 97.67% for all labels. This corresponds to the average of 96.78% in the first/second quadrant and 98.57% in the third/fourth quadrant. For performing diagnostics, this reaches scores from 88.41% to 98.17%, with an average of 96.77%. We have also provided some additional metrics, such as AUC, recall, and precision, where it makes sense.

To compare these results to a convolutional neural network (CNN), we additionally trained a CNN on our dataset. We chose not to use proprietary commercial software as a

1. We provide our code at <https://github.com/leon-dahlmeier/Topology-of-a-smile>

Table 1: Overview of classification scores. The first table shows quadrant differentiation, while the second table compares tooth classification scores in the upper and lower jaws: accuracy (in percentages), AUC in brackets and the number of features in parentheses.

(a) Scores reached for differentiating between quadrants.

Labels	Accuracy (%)	AUC	# of Features
Upper/lower Jaw	100	1	111
First/second quadrant	98.55	0.9892	1411
Third/fourth quadrant	98.93	0.9839	911

(b) Scores for tooth classification in the upper and lower jaws.

	Upper Jaw	Lower Jaw
Labels	Acc. (%) [AUC] (# of Features)	
Types of teeth	99.27 [-] (1394)	99.64 [-] (1084)
First / second incisor	98.75 [1] (111)	100 [1] (811)
Canine	100 [1] (0)	100 [1] (0)
First / second premolar	98.75 [0.975] (311)	100 [1] (611)
First / second molar	98.75 [0.9906] (511)	100 [1] (2211)

benchmark, as its substantially larger training datasets and undisclosed architectures and training procedures preclude reproducible comparisons under conditions comparable to ours. Instead, we adopted the input configuration proposed by [Esmaeilyfard et al. \(2024\)](#), using one image from each anatomical plane (axial, sagittal, and coronal) as input. Given that this architecture achieved competitive results in their study and that we began to observe signs of overfitting, we decided not to include additional slices in the CNN training process. The network consists of two convolutional blocks, each comprising a 3×3 convolution with 32 feature maps, followed by a ReLU activation and a 2×2 max pooling layer. The extracted features are flattened and passed through a fully-connected layer with 128 units and ReLU activation, with dropout applied for regularization, before a final linear layer outputs class probabilities. Input images were normalized and augmented during training with random resized cropping, Gaussian noise, and random erasing. The model was trained using the Adam optimizer with a learning rate of 10^{-3} for 500 epochs and achieved test accuracies of 70.27% for FDI labeling and 86.67% for diagnostics. Appendix A.6 summarizes the selection of the CNN baseline and the alternatives we considered.

4. Conclusion

We introduce an automated pipeline for tooth classification and diagnosis. Regions of interest (ROIs) are extracted from CBCT data using deep learning methods, while classification is performed via persistent homology. On datasets comparable to those reported in the literature, we reach average accuracy scores of 97.67% for tooth labeling and 96.77% for diagnostic tasks, outperforming a CNN trained on the same data with accuracy of 70.27% and 86.67% respectively.

Table 2: Scores reached in the diagnosis task.

Labels	Accuracy (%)	recall	precision	# of Features
Composite yes/no	98.17	0.97	0.95	1811
Impacted yes/no	100	1	1	711
Covered by bone yes/no	100	1	1	711
Attrition/nothing	99.58	0.67	0.67	1211
Filling/Retainer	100	1	1	211
Position of the filling	96.00	-	-	464
Secondary caries	95.56	0.8	1	111

Table 3: Iterated scores for all final predictions, together with overall averages.

FDI-Labels	Acc. (%)	Diagnostic Labels	Acc. (%)	recall	precision
12, 11, 21, 22	96.60	Impacted	98.17	0.97	0.95
13, 23	97.83	Covered by bone	98.17	0.97	0.95
14, 15, 24, 25	96.60	Attrition	97.75	0.67	0.67
16, 17, 26, 27	96.60	No findings	97.75	0.97	0.95
42, 41, 31, 32	98.57	Filling (with pos.)	94.24	0.98	0.94
43, 33	98.57	Secondary caries	88.44	0.8	0.94
44, 45, 34, 35	98.57	Retainer	98.17	0.97	0.95
46, 47, 36, 37	98.57				

	Our methods (%)	CNN reference method (%)
FDI-Labels	97.67	70.27
Diagnostic	96.77	86.67

The labeling and diagnosing in our workflow relies on small datasets and few features, enabling faster and less computationally demanding training. It also accommodates regions of interest of varying size.

By automating tooth classification, our approach reduces clinician workload and supports more efficient diagnostics. Future work includes increasing the sample size to further improve performance and extending the persistent homology framework to tooth detection itself, thereby eliminating the need for a separate segmentation step.

Acknowledgments

This work has received funding from the Swiss State Secretariat for Education, Research, and Innovation (SERI).

References

- Henry Adams, Tegan Emerson, Michael Kirby, Rachel Neville, Chris Peterson, Patrick Shipman, Sofya Chepushtanova, Eric Hanson, Francis Motta, and Lori Ziegelmeier. Persistence images: A stable vector representation of persistent homology. *Journal of Machine Learning Research*, 18(8):1–35, 2017.
- Aaron Adcock, Daniel Rubin, and Gunnar Carlsson. Classification of hepatic lesions using the matching metric. *Computer Vision and Image Understanding*, 121:36–42, 2014.
- Dashti Ali, Aras Asaad, Maria-Jose Jimenez, Vidit Nanda, Eduardo Paluzo-Hidalgo, and Manuel Soriano-Trigueros. A survey of vectorization methods in topological data analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):14069–14080, 2023.
- Peter Bubenik. Statistical topological data analysis using persistence landscapes. *J. Mach. Learn. Res.*, 16:77–102, 2015.
- Gunnar Carlsson. Topological pattern recognition for point cloud data. *Acta Numerica*, 23: 289–368, 2014.
- David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37(1):103–120, Jan 2007. ISSN 1432-0444. doi: 10.1007/s00454-006-1276-5.
- Z. Cui, C. Li, and W. Wang. Toothnet: automatic tooth instance segmentation and identification from cone beam ct images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6368–6377, 2019.
- Z. Cui, Y. Fang, L. Mei, et al. A fully automatic AI system for tooth and alveolar bone segmentation from cone-beam ct images. *Nature Communications*, 13:2096, 2022.
- Edelsbrunner, Letscher, and Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4):511–533, 2002.
- Rasool Esmaeilyfard, Haniyeh Bonyadifard, and Maryam Paknahad. Dental caries detection and classification in cbct images using deep learning. *International Dental Journal*, 74(2): 328–334, 2024. ISSN 0020-6539. doi: <https://doi.org/10.1016/j.identj.2023.10.003>. URL <https://www.sciencedirect.com/science/article/pii/S0020653923009358>.
- M. Ezhov, A. Zakirov, and M. Gusarev. Coarse-to-fine volumetric segmentation of teeth in cone-beam CT. In *IEEE International Symposium on Biomedical Imaging*, pages 52–56, 2019.

- R. C. Fontenele, M. D. N. Gerhardt, F. F. Picoli, A. Van Gerven, S. Nomidis, H. Willems, D. Q. Freitas, and R. Jacobs. Convolutional neural network-based automated maxillary alveolar bone segmentation on cone-beam computed tomography images. *Clinical Oral Implants Research*, 34(6):565–574, 2023.
- M. D. N. Gerhardt, R. C. Fontenele, A. F. Leite, P. Lahoud, A. Van Gerven, H. Willems, A. Smolders, T. Beznik, and R. Jacobs. Automated detection and labelling of teeth and small edentulous regions on cone-beam computed tomography using convolutional neural networks. *Journal of Dentistry*, 122:104139, 2022.
- I. E. Hamamci, S. Er, E. Simsar, A. Sekuboyina, M. Gundogar, B. Stadlinger, A. Mehl, and B. Menze. Diffusion-based hierarchical multi-label object detection to analyze panoramic dental X-rays. In H. Greenspan et al., editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, volume 14225, pages 384–392. Springer, Cham, 2023.
- Tomasz Kaczynski, Konstantin Mischaikow, and Marian Mrozek. *Computational Homology*. Springer, 2004.
- Sara Kališnik. Tropical coordinates on the space of persistence barcodes. *Foundations of Computational Mathematics*, 2018.
- S. Lee, S. Woo, J. Yu, J. Seo, J. Lee, and C. Lee. Automated cnn-based tooth segmentation in cone-beam CT for dental implant planning. *IEEE Access*, 8:50507–50518, 2020.
- Richard M. Levenson, Yashbir Singh, Bastian Rieck, Quincy A. Hathaway, Colleen Farrelly, Jennifer Rozenblit, Prateek Prasanna, Bradley Erickson, Ashok Choudhary, Gunnar Carlsson, and Deepa Sarkar. Advancing precision medicine: Algebraic topology and differential geometry in radiology and computational pathology. *Laboratory Investigation*, 104(6):102060, 2024. ISSN 0023-6837. doi: <https://doi.org/10.1016/j.labinv.2024.102060>.
- Y. Mikia, C. Muramatsu, T. Hayashi, X. Zhou, T. Hara, A. Katsumata, and H. Fujita. Classification of teeth in cone-beam CT using deep convolutional neural network. *Computers in Biology and Medicine*, 80:24–29, 2017.
- N. Morgan, J. Meeus, S. Shujaat, S. Cortellini, M. M. Bornstein, and R. Jacobs. CBCT for diagnostics, treatment planning and monitoring of sinus floor elevation procedures. *Diagnostics*, 13(10):1684, 2023.
- Stanley Nelson. *Wheeler’s Dental Anatomy, Physiology and Occlusion*. Elsevier, 2019.
- Yaopeng Peng, Hongxiao Wang, Milan Sonka, and Danny Z. Chen. Phg-net: Persistent homology guided medical image classification*. In *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 7568–7577, 2024. doi: 10.1109/WACV57701.2024.00741.
- A. Polizzi, V. Quinzi, V. Ronsivalle, P. Venezia, S. Santonocito, A. Lo Giudice, R. Leonardi, and G. Isola. Tooth automatic segmentation from cbct images: a systematic review. *Clinical Oral Investigations*, 27(7):3363–3378, Jul 2023. doi: 10.1007/s00784-023-05048-5.

- E. Shaheen, A. Leite, K. A. Alqahtani, A. Smolders, A. Van Gerven, H. Willems, and R. Jacobs. A novel deep learning system for multi-class tooth segmentation and classification on cone beam computed tomography. a validation study. *Journal of Dentistry*, 115, 2021.
- Yashbir Singh, Colleen M. Farrelly, Quincy A. Hathaway, Tim Leiner, Jaidip Jagtap, Gunnar E. Carlsson, and Bradley J. Erickson. Topological data analysis in medical imaging: current state of the art. *Insights into Imaging*, 14(1):58, Apr 2023. ISSN 1869-4101. doi: 10.1186/s13244-023-01413-w.
- Swiss National Foundation. Dentexion - the dentist’s ai assistant to objectively diagnose patients. SNF Bridge Project – Proof of Concept, 216802, 2022.
- Renata Turkeš, Jannes Nys, Tim Verdonck, and Steven Latr’e. Noise robustness of persistent homology on greyscale images, across filtrations and signatures. *PLoS ONE*, 16, 2021. URL <https://api.semanticscholar.org/CorpusID:237091859>.
- Robin Vandaele, Pritam Mukherjee, Heather Marie Selby, Rajesh Pravin Shah, and Olivier Gevaert. Topological data analysis of thoracic radiographic images shows improved radiomics-based lung tumor histology prediction. *Patterns*, 4(1):100657, 2023. ISSN 2666-3899. doi: <https://doi.org/10.1016/j.patter.2022.100657>.
- B. Xiang, J. Lu, and J. Yu. Evaluating tooth segmentation accuracy and time efficiency in cbct images using artificial intelligence: A systematic review and meta-analysis. *Journal of Dentistry*, 146:105064, 2024.
- W. Zhou, D. Zhao, B. Zhao, M. Shen, and X. Liu. A deep learning framework for automated dental segmentation and diagnostic report generation from cone-beam computed tomography. *Head & Face Medicine*, 21(1):80, Nov 28 2025. doi: 10.1186/s13005-025-00555-0.

Appendix A. First Appendix

We will now elaborate on the ablation studies we performed and the decisions we took, in order to settle on [A.1](#) the normalization method, [A.2](#) the tooth detection and its impact on the ROI, [A.3](#) the location of two dimensional slices, [A.4](#) the vectorization method, [A.5](#) the SVM structure, [A.6](#) the choice of CNN. Since the methodology provided in this section is not always the final approach, all scores should be understood relatively within the different experimentation rather than absolutely.

We started by using the three-dimensional voxel set of the whole tooth, together with five two-dimensional slices in each anatomical direction. From there we used the filtration as seen in Subsubsection [2.1.2](#).

A.1. Normalization method:

Since the images analyzed in this manuscript were not all acquired using the same machine, the ranges of grayscale values vary substantially between images. A common way to account for this variability when using SVM would be to normalize the images before classification. In [Table 4](#) we see the results of this. We normalized either the whole X-ray or just the region of interest in which the specified tooth lies. In each case, the grayscale values were rescaled to the intervals $[0, 1]$, $[0, 10,000]$, and $[0, 30,000]$.

The interval $[0, 1]$ was included because it is the most commonly used normalization range. The range $[0, 10,000]$ was chosen to investigate the effect of preserving a larger portion of the original grayscale variation, as the dynamic range differs substantially between images. Finally, the range $[0, 30,000]$ was selected because it exceeds the grayscale range observed in any image of the dataset, thereby providing a setting with minimal compression of intensity values.

Overall, the highest classification accuracy was achieved without any normalization. The way the filtrations are set up, there is exactly one feature persisting until infinity, specifically the first connected component. Since the vectorization methods cannot work with the value ‘inf’ we exchange it for the highest gray-scale-value +1 in a given image. This choice ensures that the feature still records the maximal grayscale value of the image while remaining finite, allowing it to be processed by the vectorization methods and subsequently by the SVM.

Normalization Method	Testing Accuracy (%)	Training Accuracy (%)
whole image to $[0, 1]$	57.29	100.00
tooth to $[0, 1]$	65.95	100.00
whole image to $[0, 10'000]$	86.67	100.00
tooth to $[0, 10'000]$	86.49	100.00
whole image to $[0, 30'000]$	86.31	100.00
tooth to $[0, 30'000]$	86.18	100.00
no normalization	88.29	100.00

Table 4: Training and testing accuracy for various normalization methods with Testing accuracy heat map.

A.2. Tooth detection and impact on ROI:

This aspect warrants further investigation. In the present work, we adopted the segmentation-based approach in order to obtain a fully automatic workflow.

The exact impact of suboptimal selected ROI’s still needs to be further investigated. But we have run our tests twice, once using the segmentation approach described above and once using manually selected ROI’s, ensuring voxel level precision.

- To reiterate the results when using the **segmentation approach**: In the FDI-labeling task, this results in scores ranging from 96.60% to 98.57% averaging 97.67% for all labels. For performing diagnostics, this reaches scores from 88.41% to 98.17%, with an average of 96.77%.
- And in comparison the **manually selected ROI’s**: For the FDI-labeling task, we reached scores ranging from 97.31% to 98.55% averaging at 97.89%. For performing diagnostics, we reached scores from 90.96% to 99.13% averaging at 96.77%.

A.3. Slice location:

We further investigated the impact of changing the locations of the two-dimensional slices. To this end, we fixed one slice at the midpoint of each dimension and varied the positions of the remaining four slices. Using the parameters ‘inner’ and ‘outer’, we decided the position of the remaining slices as follows: the two slices closest to the slice in the middle were placed at position $(10 \pm \text{inner}) \cdot (\text{dimension}/20)$, while the outermost slices were placed at $(\text{outer}) \cdot (\text{dimension}/20)$ and $(20 - \text{outer}) \cdot (\text{dimension}/20)$. The resulting performance is reported in Table 5. We adopted the best-performing configuration for all subsequent experiments.

inner \ outer	1	2	3	4	5
1	78.38%	83.78%	82.88%	84.68%	82.88%
2	81.08%	82.88%	83.78%	87.39%	84.68%
3	81.08%	86.49%	88.29%	85.59%	83.78%
4	77.48%	82.88%	83.78%	80.18%	82.88%
5	75.68%	83.78%	85.59%	83.78%	81.98%

Table 5: Different locations for the two-dimensional slices with heat map.

A.4. Vectorization method:

Using the code provided by (Ali et al., 2023), we compared a range of vectorization methods to determine the one best suited for our task. Descriptions of the methods not discussed in Subsubsection 2.1.4 can be found in that reference. In Table 6 we can see that extracting statistical features generated the highest accuracy. We therefore adopted this vectorization method for all subsequent experiments.

Vectorization Method	Testing Accuracy (%)	Training Accuracy (%)
PersStats	88.29	100.00
AlgebraicFunctions	75.68	100.00
BettiCurveFeature	71.17	100.00
ComplexPolynomialFeature	74.77	100.00
PersImageFeature	34.35	100.00
PersLandscapeFeature	87.39	100.00
PersLifespanFeature	78.38	100.00
PersSilhouetteFeature	76.58	100.00
PersTropicalCoordinatesFeature	82.88	100.00

Table 6: Training and testing accuracy for various vectorization methods with testing accuracy heat map.

A.5. The SVM structure:

So far we used a grid search to find the best kernel for the SVM, including polynomial, radial basis function (RBF), and sigmoid kernels. A linear kernel which $C = 0.001$ consistently performed the best. We therefore settled on skipping the grid search for all following iterations, in order to combat overfitting.

Additional features such as Texture analysis did not improve test accuracy.

The tests described up until now including this one the three-dimensional voxel set of the whole ROI, together with five two-dimensional slices in each anatomical direction. In order to combat overfitting, as seen in Table 6 and to understand which features are the most important, we reduced the number of filtrations. In Table 7 we see on the left what remains, that is, whether or not the three-dimensional image was used, and from the two-dimensional image we differentiate between the coronal sagittal and axial slices as well as the homology of dimension one and zero. We ran these tests for both the statistical and the landscape-feature approach to vectorization. We primarily focused on reducing to the two-dimensional images, from which we also either extracted only one- or zero dimensional homology. We expected dimension 1 homology to be the more important feature compared to dimension 0. We therefore additionally started by only using dimension one homology of the two-dimensional images, but further left out the three-dimensional images or the two-dimensional slices with either even or odd index. We want to point out the row using all features previously used, resulting in the best accuracy. Leaving out the three-dimensional region of interest only slightly reduced the overall accuracy.

We therefore settled on not manually reducing our features but rather using a more methodical approach, akin to principal component analysis (PCA). Specifically we train a linear SVM with $C = 0.001$ and order the features by their assigned weights. We then only use the features with the highest weights to perform 5-fold cross-validation, iteratively increasing the number of used features.

3D	d1 coronal	d1 sagital	d1 axial	d0 coronal	d0 sagital	d0 axial	Stat. acc. (%)	Ladsc. acc. (%)
✓	✓			✓			74.77	64.32
✓		✓			✓		75.14	64.14
✓			✓			✓	75.86	75.32
✓	✓	✓	✓				86.13	78.92
	✓	✓	✓				86.31	77.66
✓	✓	✓	✓	✓	✓	✓	87.57	81.26
✓				✓	✓	✓	87.75	74.41

Table 7: Test and train accuracy scores for fewer two-dimensional images with heat map shading.

A.6. The choice of CNN:

We decided against using proprietary commercial software as a baseline, since such systems are typically trained on substantially larger datasets and their architectures and training procedures are not publicly available. As a result, they cannot be reproduced or trained under conditions comparable to our own.

We therefore considered two different approaches:

1. Using the pre-trained ResNet50, we generate vectors for three axial slices of the tooth. These vectors were concatenated and used as data to train and test an MLP resulting in an accuracy of 82.88% for FDI labeling and 78.87% for diagnostics. Unfortunately, ResNet50 is trained on a bigger dataset than what we worked with and is not trained specifically for teeth making it harder to compare with our approach.
2. We settled on using the approach described in (Esmailyfard et al., 2024). The advantages are twofold:
 - The architecture is publicly available, allowing us to implement and train the network ourselves rather than relying on a proprietary black-box system.
 - The network is trained exclusively on our dataset, ensuring a fair comparison with our topological approach, since both methods use the same training data.