Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

Dataset Annotation Tools for Data-driven Segmentation of Structured Point Clouds

Bachelor Thesis

2024
Matúš Kočalka

Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

Dataset Annotation Tools for Data-driven Segmentation of Structured Point Clouds

Bachelor Thesis

Study Programme:	Computer Science
Field of Study:	Computer Science
Department:	Katedra aplikovanej informatiky FMFI UK, Mlynská dolina, 842 48 Bratis
Supervisor:	doc. RNDr. Martin Madaras, PhD.

Bratislava, 2024
Matúš Kočalka

Acknowledgments: Tu môžete poďakovať školiteľovi, prípadne ďalším osobám, ktoré vám s prácou nejako pomohli, poradili, poskytli dáta a podobne.

Contents

List of Figures

List of Tables

Introduction

Segmentation represents a cornerstone challenge in computer vision. It finds application almost everywhere where a 2D image or 3D scene is processed in some way.

However, despite its fundamental importance, segmentation poses an inherently complex problem. And Fortunately in recent years, deep learning was founded as attractive solution, that provides a steady results. Yet, this solution is not without its drawbacks, notably its reliance on training data, i.e., annotated datasets.

These annotated datasets are typically manually labeled data, requiring some human input for their creation. In response to this need, many tools have been developed to assist users in annotation training data and thus reduce time required for their creation.

Motivation

When it comes to 2D segmentation or classification, there are wealth of freely available annotated datasets and accessible tools capable of annotating un-annotated 2D datasets, which are also widely accessible.

However, when it comes to 3D data, it becomes notably more challenging. One of the issues is the scarcity of published 3D datasets, often due to proprietary concerns or sheer volumetric complexity. And secondly, although many tools have been developed for 3D annotation, their utilization proves more intricate and time-consuming compared to their 2D counterparts.

Nevertheless, a subset of 3D data, known as structured point clouds, which have the property of being structured in a 2D grid can therefore be represented as 2D images. Therefore, they can be annotated in various tools designed for annotating 2D images. However, structured point clouds contain much more information than typical RGB images and traditional 2D annotation tools do not take this into account, thus not utilizing their full potential.

Therefore, we have decided to develop a new annotation tool tailored specifically for annotating structured point clouds to achieve new and possibly more efficient way of annotating 3D training datasets.

Chapter 1

Related work

In this chapter, we will go through various existing annotation tools. We will cover the types of data they are applied to, their advantages, and disadvantages.

1.1 Annotation tools

Annotation of a document generally refers to the augmentation of additional information at any level within the document. The document can be various data formats, ranging from 2D images to sound recordings or text. Then the annotation of such document can manifest in various forms, such as, marking bounding boxes of objects in a 2D image, or directly segmenting image and describing individual objects within segmentation, or determining time intervals when spoken words or other targeted sounds occur in an audio recording.

The precision of these added pieces of information, annotations, is crucial, as the effectiveness of artificial intelligence models trained on them hinges directly upon their accuracy.

Annotation tools are typically evaluated based on two primary metrics: annotation accuracy and ease of use, i.e., user interface (UI), as their task is to mitigate and speed up the anotation of new, accurate datasets.

Annotation tools can be standard applications, web applications, or libraries of a programming language.

Within this chapter, our focus will be on a subset of annotation tools closely aligned with our research objectives. Specifically, we will delve into tools designated for annotating and segmenting 2D images, 3D scenes, and multispectral images.

1.1.1 2D images annotation tools

The fundamental techniques for labeling distinct regions within an image primarily involve manual methods, such as brush, drawing bounding boxes and polygons. How-

ever, these conventional approaches are labor-intensive and time-consuming. What's more, they often lack precision, especially in the case of bounding boxes and polygons, which offer only limited accuracy. Therefore, numerous analytical methods are used to semi-automate segmentation. These methods include thresholding, edge detection, region growing, and graph-based segmentation. Alternatively, some approaches leverage pretrained artificial intelligence models, such as SAM[11], to streamline and enhance the segmentation process.

Presently, some of the most renowned 2D annotation tools are highlighted in sources such as [13] and [14]. For instance, [13] delivers a apt overview of the topic of annotation tools designated for 2D images, coupled with an in-depth examination encompassing more than 20 widely-used tools. Afterwards, we present a few examples along with their brief descriptions.

1. Labelbox [5] One of the most commonly used annotation tool. Labelbox utilizes all the basic annotation approaches like polygons, bounding boxes, lines, as well as more advanced segmentation tools. Labelbox is deployed with open source front end and uses python API for segmentation methods

2. CVAT [2] Computer Vision Annotation Tool is the industry-leading data engine for machine learning. CVAT is specialized in annotating videos and images annotation for Computer Vision algorithms.

3. LabelMe [4] Another online annotation tool to build image databases for computer vision research. LabelMe was created by the MIT Computer Science and Artificial Intelligence Laboratory originally in python and QT for frontend.

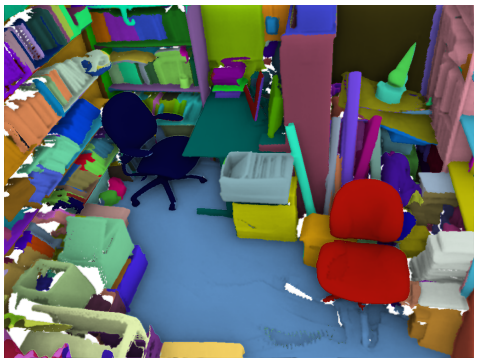
4. Microsoft VoTT [8] An open source annotation and labeling tool for image and video assets. VoTT is a React + Redux Web application, written in TypeScript. Also provides import and export custom data to local or cloud storage providers.

5. OpenCv [9] While not a dedicated annotation tool, OpenCV reigns as the world's largest computer vision library, and is often integrated in the backend of annotation tools to perform image segmentation.

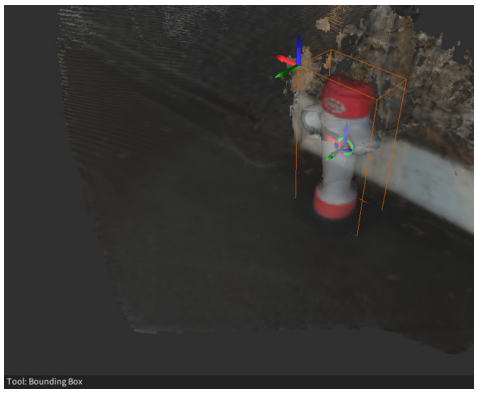
1.1.2 3D images annotation tools

An annotation tool designed for 3D data presents several unique challenges when compared to traditional 2D data annotation.

[1] `https://github.com/strayrobots/3d-annotation-tool/tree/main`
[2] `https://hkust-vgd.ust.hk/scenenn/home/iros18/index.html`



(a) [1]



(b) [2]

Figure 1.1: Example of 3D annotation tools. (a) Final segmentation with A Robust 3D-2D Interactive Tool for Scene Segmentation and Annotation [15]. (b) Placement of 3D bounding box inside 3D annotation tool by Kenneth Blomqvist [1]

Firstly, 3D scenes inherently contain more information than 2D images, theoretically allowing for more precise segmentation. However, this also necessitates the development of new segmentation methods tailored specifically for 3D scenes, or the extension of existing 2D segmentation techniques to accommodate an additional dimension.

Another significant challenge in 3D annotation is based on fact that a 2D computer screen is not ideally suited for working with 3 dimensional data. To address this issue, various approaches have been devised.

One such approach involves annotating a 2D projection of the 3D scene from a specific viewpoint. This method allows for familiar annotation tools like brushes, bounding boxes, and polygon labeling, just like to those used in 2D image annotation. Additionally, by incorporating additional information such as point distances or depth (point distance from the camera), higher-quality segmentations can be achieved compared to traditional 2D methods. An example of such an approach is RGB-D segmentation [16], where superpixel segmentation is used to distinguish areas with similar color information based on 3D geometric reconstruction.

However, this approach may not be suitable for all scenarios, particularly for 3D scenes constructed from multiple viewpoints, where such segmentation made on projection from single view point may be incomplete or inaccurate.

Another approach is to abandon the 2D concept and annotate datasets directly within a 3D environment using specialized 3D viewers or virtual reality. While more practical and reliable for multi-viewpoint 3D scenes, this approach introduces complexity, especially for large scenes comprising millions of points. Just displaying the entire scene alone can be challenging. Additionally annotation in 3D space is also not as straightforward as with 2D images and can by confusing for users. Manual annota-

tion tools for 3D data typically offer features such as placing 3D bounding boxes and brushes.

Kenneth Blomqvist offers such a 3D annotation tool [1], which includes annotation using 3D bounding boxes, key points and squares to mark areas. Annotation using a 3D bounding box can be seen in Figure 1.1(b).

Another excellent example is "A Robust 3D-2D Interactive Tool for Scene Segmentation and Annotation" [15] by Duc Thanh Nguyen and others. In theyr work is introduced a great annotation tool for 3D scenes with the ability to annotate both 2D projections and 3D scenes simultaneously. They also developed assistive, interactive operations such as undo, split, extract, and merge to handle automatic initial segmentation, along with automatic tools for segmentation and object search. Examples of annotations created using these tools are depicted in Figure 1.1(a).

1.1.3 Multispectral Images Annotation tools

A multispectral image is a collection of multiple 2D images, or channels, where each of these channels represents the intensity value of a specific wavelength of the electromagnetic spectrum from the same scene [17]. Multispectral images are captured using special sensors and find widespread applications, particularly in agriculture, where they aid in field monitoring and differentiation between healthy and diseased crops.

Annotation of these data also poses its own set of challenges. One major challenge is the time-intensive nature of the process, as if we have a dataset containing M images, each with N channels, annotating them would require annotating $M \times N$ images. It makes much more sense to annotate individual multispectral images as a whole rather than their individual channels separately, as all of them are derived from the same scene. However, annotating multispectral images as a whole requires considering all channels simultaneously, which complicates segmentation and annotation.

Supervisely [10] [3] is an excellent annotation tool for annotation and segmentation of multispectral images, as well as 3D scenes or videos. It offers a comprehensive set of classic annotation tools and integrates advanced artificial intelligence tools, such as SAM [11].

Chapter 2

Proposal

Annotation of 3D data can be complicated, and even with sophisticated tools, complex scenes often requiring significant time and effort to annotate.

However, for one subgroup of 3D data, a different approach can theoretically be chosen. This subgroup consists of structured point clouds. These are 3D data organized in a 2D grid, and therefore allowing an injective mapping to 2D space. Consequently, if we focus solely on the texture of such a point clouds, they could be annotated using conventional 2D annotation tool with 2D segmentation.

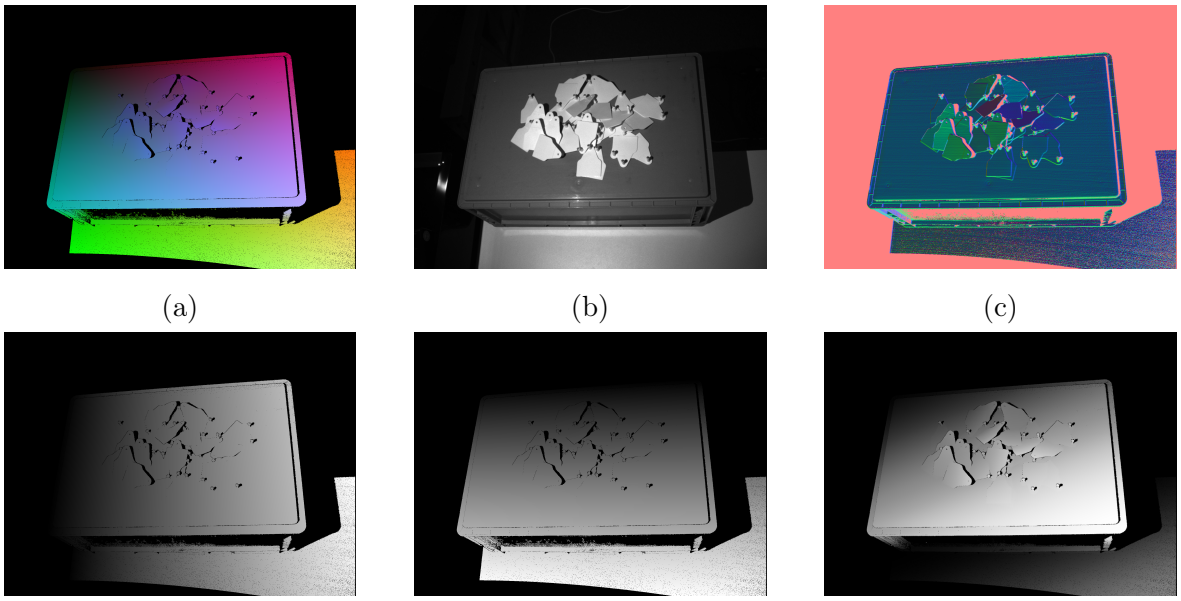
In our work, we introduce a new approach to annotating structured point clouds, leveraging 2D methods to achieve a simple interface for fast 3D data annotation. Beyond traditional annotation techniques like brushes and bounding boxes, we aim to to implement automatic segmentation. This automated segmentation serves as the initial state of annotation, which users can refine manually in subsequent steps. Subsequently after user fully annotate firs scene, we intend to use this segmentation as model to evaluate the most suitable segmentation method for this scene and thus for remaining scenes in the dataset, as we assume that images from one dataset come from a similar scene. In this way, we aim to iteratively enhances the efficiency of the initial automatic segmentation and thus simplifies the annotation of the dataset for the user as much as possible.

This chapter delineates the precise scope of the problem we tackle in our research along with the the individual challenges we had to solve in its solution.

2.1 Structured point clouds

Let's start by quote of an introduction to point clouds from MatLab documentation [7]:

> There are two types of point clouds: organized and unorganized. These describe point cloud data stored in a structured manner or in an arbitrary



(a) (b) (c)

(d) (e) (f)

Figure 2.1: Example of structured point cloud. We can see individual channels. (a) Points coordination. Red color represents x coordination, green color represents y coordination and finally z coordination as blue color. (b) Intensity texture channel, which contains color intensities of scene. (c) Normal's map channel, for each point cloud point is recorded its surface normal vector. (d) Separate x coordination values (e) Separate y coordination values (f) Depth map channel, per point distance from camera.

> fashion, respectively. An organized point cloud resembles a 2-D matrix, with its data divided into rows and columns. The data is divided according to the spatial relationships between the points. As a result, the memory layout of an organized point cloud relates to the spatial layout represented by the xyz-coordinates of its points. In contrast, unorganized point clouds consist of a single stream of 3-D coordinates, each coordinate representing a single point. You can also differentiate these point clouds based on the shape of their data. Organized point clouds are M-by-N-by-3 arrays, with the three channels representing the x-, y-, and z- coordinates of the points. Unorganized point clouds are M-by-3 matrices, where M is the total number of points in the point cloud.
>
> Most deep learning segmentation networks, such as SqueezeSegv1/v2, RangeNet++, and SalsaNext, process only organized point clouds. In addition, organized point clouds are used in ground plane extraction and key point detection methods. This makes organized point cloud conversion an important pre-processing step for many Lidar Toolbox™ workflows.

As highlighted in the preceding quote, structured point clouds have been employed in various applications. In our research we are also particularly interested in structured point clouds, due to their property of having injective mapping to 2D space.

We can see example of injective mapping of structured point cloud to 2D in figure 2.1. Typically structured point clouds bare more information than just x, y, z coordinations. they usually contains also surface normals, texture and depth map. Each one of these type of data can by represented in separate image, channel. This also can be seen in figure 2.1. In each channel is used the same injective mapping, or structure, but with different entry.

Structured point clouds can be acquired through various means. In the following subsections, we will delve into some of these acquisition methods.

2.1.1 Conversion of unorganized point clouds

Unorganized point clouds are a common output from 3D scanners. Their structure consists of a long stream of individual x, y, z coordinates, or multiple streams if the point cloud contains additional data such as normals or intensities. In certain scenarios, such unorganized point clouds can be organized into a 2D grid by leveraging stereo dependencies among individual points, thereby converting them into a structured point cloud.

The Matlab documentation provides an illustrative example of this conversion process. Specifically, it demonstrates conversion using spherical projection from unorganized point clouds produced by lidar sensors [6]. However to enable this conversion it

is necessary to provide the parameters of the lidar sensor under which the given point cloud was captured.

2.1.2 Synthetic structured point clouds generation

Another way to acquire structured point clouds is through synthetic generation. As mentioned earlier, the Achilles' heel of neural networks lies in their dependence on high-quality and comprehensive training datasets. With respect to 3D data, creating such datasets from real scenes using 3D sensors and subsequently annotating them can be challenging and time-consuming.

Hence, a common practice is to generate synthetic datasets for training artificial intelligence models. This approach bypasses both the acquisition and annotation processes. Since annotation is also automatically generated with the scene, as the content and locations within the generated scene are predetermined. However, the primary challenge of this pipeline lies in ensuring the realism and diversity of the resulting data. Creating synthetic data that closely resembles real-world scenarios is essential, as the performance of trained artificial intelligence models on real data will depend on it.

An example of such a pipeline is the work of Peťo Kravár [12]. In this work, he introduces the SynBin pipeline for generating synthetic bins, as he states in his paper:

> As a solution to the outlined issues, this thesis aims to propose SynBin, a scalable rendering pipeline for synthetic scans of bins. The pipeline contains parametric bin models able to create large amounts of mutually different bins sharing similar traits. Additionally, the pipeline provides an option to add other imported objects to the scene. The produced datasets contain sets of synthetically rendered structured point clouds of a bin model in digital environment and ground truth data.

2.1.3 Structured point clouds as a output of 3D scanner

Some 3D scanners offer structured point clouds directly as output. An exemplar of such technology is the PhoXi 3D scanner developed by Photoneo[1]. In the scope of our research, we worked with scans obtained directly from such cameras. Figure 2.1 illustrates a structured point cloud acquired using this method.

To capture 3D information of a scene, PhoXi cameras utilizes structured light technology. These cameras are equipped with both a projector and a camera. During scanning, the projector projects light patterns onto the scene, which are then captured

[1] https://www.photoneo.com/phoxi-3d-scanner/

by the camera. By analyzing the deformations of these patterns on the surface of the scene, the software is able to reconstruct its detailed three-dimensional representation.

Bibliography

[1] 3d annotation tool by kenneth blomqvist. [Citované 2024-04-26] Dostupné z https://github.com/strayrobots/3d-annotation-tool.

[2] Cvat home page. [Citované 2024-04-26] Dostupné z https://www.cvat.ai/.

[3] How to annotate multispectral images for computer vision models. [Citované 2024-04-28] Dostupné z https://supervisely.com/blog/labeling-multispectral-images/.

[4] Labelme home page. [Citované 2024-04-26] Dostupné z http://labelme.csail.mit.edu/Release3.0/.

[5] Lebelbox home page. [Citované 2024-04-26] Dostupné z https://labelbox.com/.

[6] *Mathlab documentation. Unorganized to Organized Conversion of Point Clouds Using Spherical Projection.* [Citované 2024-05-8] Dostupné z https://www.mathworks.com/help/lidar/ug/unorgaized-to-organized-pointcloud-conversion.html#mw_rtc_UnorganizedToOrganizedConversionOfPointCloudsExample_M_D1BDC4DA.

[7] *Mathlab documentation. What are Organized and Unorganized Point Clouds?* [Citované 2024-05-8] Dostupné z https://www.mathworks.com/help/lidar/gs/organized-and-unorganized-point-clouds.html.

[8] Microsoft wott git repository. [Citované 2024-04-26] Dostupné z https://github.com/microsoft/VoTT.

[9] opencv main page. [Citované 2024-04-26] Dostupné z https://opencv.org/.

[10] Supervisely, platform that integrates countless open-source tools and custom built solutions. [Citované 2024-04-28] Dostupné z https://supervisely.com/.

[11] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. Segment anything. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3992–4003, 2023.

[12] Peter Kravár. Synthetic dataset rendering of 3d scans for robust 6d bin pose estimation [online]. Master's thesis, Brno University of Technology, Faculty of Information Technology, 2023 [cit. 2024-05-11]. Supervisor Mgr. Matúš Talčík.

[13] Anisha Rebinth and Mohan Kumar S. Importance of manual image annotation tools and free datasets for medical research. *Journal of Advanced Research in Dynamical and Control Systems*, 10:1880–1885, 01 2019.

[14] Alberto Rizzoli. 13 best image annotation tools of 2023 [reviewed]. discover 13 most popular image annotation tools of 2023. compare their features and pricing, and choose the best data annotation tool for your needs, 2013. [Citované 2024-04-26] Dostupné z `https://www.v7labs.com/blog/best-image-annotation-tools`.

[15] Duc Thanh Nguyen, Binh-Son Hua, Lap-Fai Yu, and Sai-Kit Yeung. A robust 3d-2d interactive tool for scene segmentation and annotation. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 2017.

[16] Jingyu Yang, Ziqiao Gan, Xiaolei Gui, Kun Li, and Chunping Hou. 3-d geometry enhanced superpixels for rgb-d data. In Benoit Huet, Chong-Wah Ngo, Jinhui Tang, Zhi-Hua Zhou, Alexander G. Hauptmann, and Shuicheng Yan, editors, *Advances in Multimedia Information Processing – PCM 2013*, pages 35–46, Cham, 2013. Springer International Publishing.

[17] Li You. Multispectral image processing. Master's thesis, Brno University of Technology, Faculty of Information Technology, 2021. Supervisor prof. Dr. Ing. Pavel Zemcík, Ph.D.