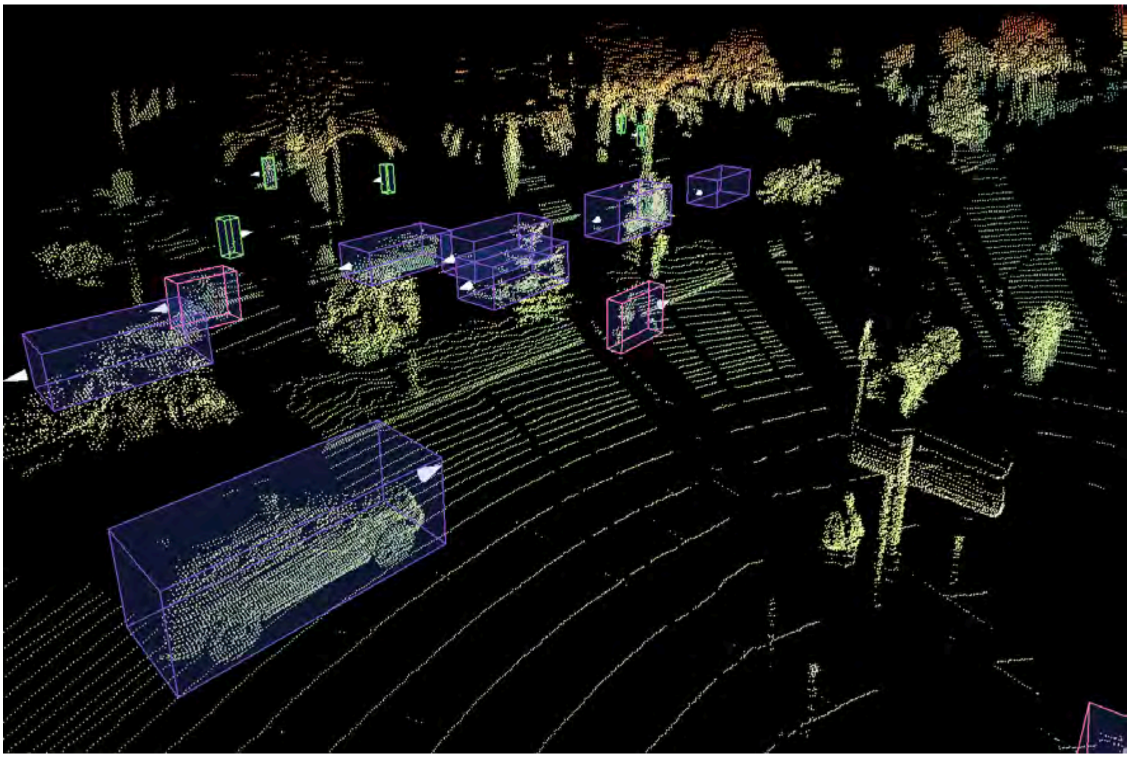


Micro-Solutions for Micro-Adjustments

Reducing Micro-Adjustments During Annotation for Data Labeling Platforms: An Interaction Design Approach

Master's Thesis in Computer Science and Engineering

Jakob Kitzing & Mats Kullerstrand

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2024

Master's thesis 2024

Micro-Solutions for Micro-Adjustments

Reducing Micro-Adjustments During Annotation for Data Labeling Platforms: An Interaction Design Approach

JAKOB KITZING & MATS KULLERSTRAND

UNIVERSITY OF GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2024

A Chalmers University of Technology Master's thesis template for LaTeX
Reducing Micro-Adjustments During Annotation for Data Labeling Platforms: An Interaction Design Approach
Jakob Kitzing & Mats Kullerstrand

Supervisor: Beata Stahre Wästberg, Computer Science and Engineering
Advisor: Jessica Karlsson & Oskar Samuelsson, Kognic
Examiner: Morten Fjeld, Computer Science and Engineering

Master's Thesis 2024
Department of Computer Science and Engineering
Chalmers University of Technology and University
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Image of 3D point cloud captured by lidar. Screenshot provided with consent from Kognic.

Typeset in LaTeX
Gothenburg, Sweden 2024

A Chalmers University of Technology Master's thesis template for $\LaTeX$
Reducing Micro-Adjustments During Annotation for Data Labeling Platforms: An Interaction Design Approach
Jakob Kitzing & Mats Kullerstrand
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

The autonomous vehicles sector has brought computerization into driving, a task that previously was solely for humans. This results in companies providing software and annotation services for machine learning, helping with labeling, categorizing, and tracking data. This Master's thesis was conducted at one of these annotation companies, Kognic.

The act of making adjustments to annotations of already sufficient quality, called micro-adjustments, is one optimization problem existing in Kognic's annotation platform. However, annotations still need expert-level quality to generate as close as possible to 'ground truth', a reality for the machine learning model to train on. Since the process handles huge datasets, the time spent on micro-adjustments results in major time losses. The thesis goal is to analyze annotator workflows, propose one or multiple solutions to improve the way annotators work with object annotation, and create general design guidelines for the reduction of micro-adjustments in annotation platforms. This was achieved by utilizing interaction design following the research question:

In which aspects can interaction design support the reduction of micro-adjustments (small repetitive unnecessary adjustments of annotations of already sufficient quality) in annotations of 3D point clouds?

The study utilized aspects such as design thinking, user-centered design, and the triple diamond design process to address the wicked problem. The thesis showed that the iterative nature of interaction design is an efficient approach when designing to reduce micro-adjustments. The result was three final design solutions, Line Assist, Two Pointy, and Sequence Process Wizard, and 11 design guidelines.

The design guidelines contribute to the field of annotation by outlining important considerations when designing for the reduction of micro-adjustments. These guidelines emphasize the importance of rotation micro-adjustments, feedback to annotators, improving existing tools over creating new ones, annotators culture, and how micro-solutions can minimize micro-adjustments. Additionally, the guidelines can assist in projects with users located far away from where the study takes place, highlighting the significance of gathering information from multiple sources.

Keywords: Annotation, Data Labeling, Autonomous Vehicles, Micro-Adjustments, 3D Point Cloud, 3D Platform, User Experience, Interaction Design, User-Centered Design, HCI.

Acknowledgements

We would like to thank our mentors at Kognic, Jessica Karlsson and Oskar Samuelsson for being a great support and engaged in helping us succeed throughout the project. We would also like to express our gratitude to the people at Kognic, where each person we approached helped us without hesitation, with everything from workshops to general questions. Additionally, a special thank you to the annotators who participated in the observations, interviews, A/B testing, and surveys.

Lastly, we would like to thank our academic supervisor at Chalmers University of Technology, Beata Stahre Wästberg, for providing valuable feedback and encouragement throughout the project.

Jakob Kitzing & Mats Kullerstrand, Gothenburg, 2024-06-10

Contents

List of Abbreviations

Abbreviation	Definition
AV	Autonomous Vehicle
ML	Machine Learning
DL	Deep Learning
UX	User Experience
UI	User Interface
HCI	Human-Computer Interaction
UCD	User-Centered Design
KLM	Keystroke Level Model
Hi-Fi	High-Fidelity
Lo-Fi	Low-Fidelity
BPO	Business Process Outsourcing

1
Introduction

Autonomous vehicles (AVs) is a relatively new sector in the automobile and technology industries. Through remarkable improvements, this sector has brought computerization into driving, which previously was a task solely for humans. [1]

Kognic specializes in providing software and annotation services for *machine learning* (ML) in the autonomous vehicle sector, helping companies label, categorize, and track data. Their mission is aligning data for embodied AI with their software tool, the *Kognic Platform*, an annotation and visualization tool-set for computer vision [2]. Annotations are the foundation of the ML models for autonomous vehicles, where the labeled datasets include metadata that is used to teach the models to work as intended. Annotators work with marking out and labeling objects; through iterations, they achieve high-quality annotations.

Achieving a good cost-to-quality ratio is a key aspect when creating large datasets of labeled data. One existing optimization problem for Kognic is the efficiency during annotations and why annotators adjust annotations of already sufficient quality, called *micro-adjustments*. Since the process handles huge datasets, small changes can make a major impact on how much time is spent on each task. This Master's thesis will investigate the possibility of reducing unnecessary micro-adjustments, potentially having a substantial impact on the overall efficiency of the process.

1.1 Aim and Goal

This Master's thesis will investigate when and why annotators are making micro-adjustments to annotations of already sufficient quality. The thesis aims to solve this problem by increasing annotator efficiency through design. The goal of the thesis is to analyze current annotator workflows, propose one or multiple solutions to improve the way annotators work with object annotation, and create general design guidelines for the reduction of micro-adjustments in annotation platforms.

1.2 Research Question

In which aspects can interaction design support the reduction of micro-adjustments (small repetitive unnecessary adjustments of annotations of already sufficient quality) in annotations of 3D point clouds?

1.3 Scope and Limitations

Annotators work with several different types of projects and datasets, many with different specifications of what should be included in the annotations. Since the problem area is large, the scope of this Master's thesis needs to be narrowed down to allow for adequate time on a more specified subject. At the start, the project's focus was on single-frame 3D point cloud with object annotation of cuboids, which later changed to include sequenced annotation as well.

The annotators are based outside of Sweden and have varying degrees of technical literacy in the field of ML and AVs. This perspective is to be incorporated into the final design solution. Their location also makes it difficult for this Master's thesis to conduct user research and testing in person. Therefore, these phases of the project will utilize video recordings of the annotators' workflow, digital meetings for interviews, and other user research methods. This project will not include how annotators interact with physical tools while annotating in the Kognic Platform. This limitation is set due to the annotator's location, where it is neither feasible nor possible to collect data regarding this subject.

1.4 Stakeholders

The stakeholders of this Master's thesis are Kognic, as they are the company for which this project is being undertaken, Chalmers University of Technology, as they are responsible for evaluating the project, and the thesis students, as they are conducting the project and a requirement for their Master's degree.

2

Background

This chapter presents AVs and how they operate through automotive intelligence. It also presents the company, Kognic, and its part in this industry, aligning data for embodied AI.

2.1 Autonomous Vehicles

AVs is a relatively new sector in the automobile and technology industries [1]. Through remarkable improvements, this sector has brought computerization into driving, which previously was a task solely for humans [1]. There are various levels of automation in vehicles, where *SAE On-Road Automated Driving (ORAD) Committee* provides a detailed taxonomy with definitions, ranging from no driving automation, Level 0, to full driving automation, Level 5 [3]. In this range, Level 5 would be considered fully autonomous [4].

AVs is an ongoing topic within the urban mobility sector, with the potential to revolutionize the industry in aspects such as societal benefits, money savings, lowering emissions, and road safety [1], [5]. With the assumption that by 2025, 5-20% of all driving will be autonomous, or semi-autonomous, the estimated global economic impact would be $200 billion to $1.9 trillion per year [1], [6, p. 78]. *The National Highway Traffic Safety Administration*, based on a survey spanning from 2005 to 2007, determined that 94 percent ($\pm$2.2%) of all incidents were attributed to the driver [7]. The incidents' primary causes were driver recognition, decision, performance, and non-performance errors [7]. However, there are still concerns regarding how reliable and safe AVs are through all automation levels and driving circumstances, which needs to be addressed before a universal implementation [8], [9].

The question remains if the general public is willing to switch from regular vehicles to AVs, with public acceptance being the general solution to AVs diffusion [10]. Some of the major factors affecting public acceptance are data security, personal safety, marginal cost, mobility, environmentally-friendly, and ride comfort, which can all have either positive or negative effects [10]. According to Haboucha et al. [11], there is still a large uncertainty for drivers to switch to AVs or shared AVs, where 44% would remain with their regular vehicles. Another survey performed by Othman [12], showed that safety is the main aspect of AV acceptance, where 74% of participants believed the normal driver can perform better than the AV. Fully autonomous driving is still a technology yet to be broadly implemented in society, however, there is still a great interest by the public and an increase in development by the industries [13].

2.2 Autonomous Intelligence

The AV software system uses three core categories, *perception*, *planning*, and *control* [14]. These core competencies interact with each other, the hardware, and the environment to make it possible for the vehicle to be autonomous [14], see Figure 2.1. Additionally, this software system needs vast datasets with metadata tags that decipher the relevant elements to train the ML model [15, pp. 1-2]. Below, all the concepts of the AV software system are briefly explained.

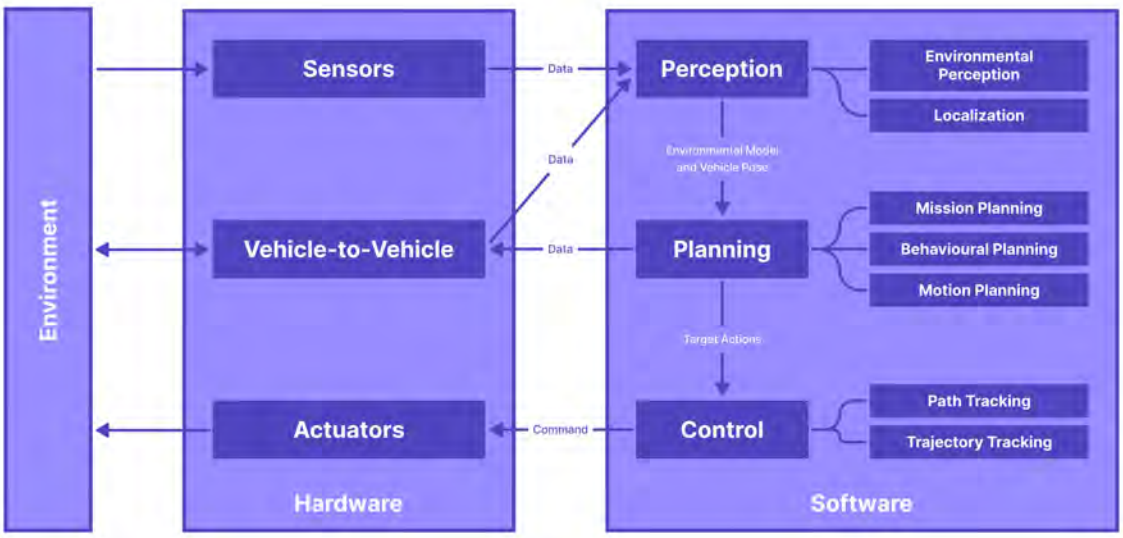


Figure 2.1: How the AVs software system works and interacts with the hardware and environment. Figure based on Pendleton et al. [14]

.

2.2.1 Perception System

For AVs to work as intended, the vehicle must have sufficient perception of its surroundings [4]. The perception system of AVs describes how well it can gather information from the surrounding environment and extract relevant knowledge [14]. In today's AVs, this perception is developed by employing ML and *deep learning* (DL) [4]. The difference between these terms is that DL is a sub-category of ML, employing artificial neural networks in an attempt to replicate the learning processes of the human brain [16]. DL models outperform ML algorithms in dataset processing and have shown superhuman capabilities in closed environments [17]. However, to enhance the performance of DL models it is crucial to provide large amounts of data, unlike traditional ML algorithms that can achieve satisfactory results with a smaller dataset [17].

The training of AVs includes instructing the vehicle to interpret its surroundings and determine appropriate behavior for various situations. The perception system of AVs relies on an aggregation of cameras, radar, and lidar sensors [18]. The latter, lidar sensors, have been proven to significantly enhance the field due to their superiority in ranging accuracy and performance in dark environments. [18], [19]. Lidar sensors utilize lasers to illuminate the surroundings and output a 3D point cloud, a visual representation of the corresponding environment [18], see Figure 2.2. For the perception system to work as efficiently as possible, lidar sensors are utilized in combination with cameras, which are well-suited for object recognition [18], [20]. The use of light-, sound-, and/or vision-based sensors is what makes today's AVs perceive their surroundings, which is important for their safety and implementation in society [21].

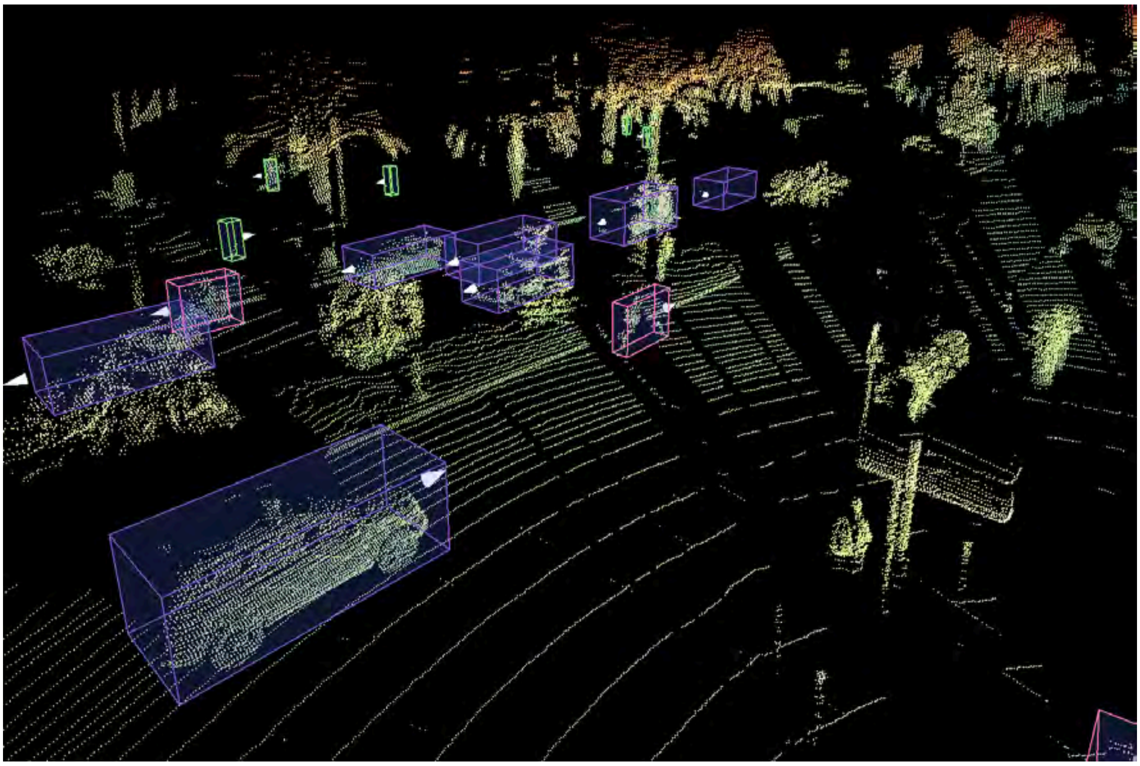


Figure 2.2: Image of 3D point cloud captured by lidar radars. Screenshot provided with consent from Kognic.

2.2.2 Planning System

The planning competency of the system accomplishes the higher-order goals of AVs. It is an intentional decision-making process divided into a three-level hierarchical framework, *mission planner*, *behavioral planner*, and *motion planner*. The mission planner considers assignments such as route decision-making. The behavioral planner engages with other agents and makes decisions when necessary, for example, road rules, lane changes, and overtakes. The motion planner decides the actions necessary to achieve the goal, such as navigation from an initial point to a destination while avoiding obstacles. [14]

2.2.3 Control System

The final core category for the software of AVs is the control system, the process of executing the intentions into actions. This is done by converting the intentions of the system into the actuators of AVs such as the steering, gas pedal, and brake pedal. It is from these hardware-level measurements one can determine the efficiency and accuracy of the system. [14]

2.2.4 Optimizing Prediction Model

ML is a general-purpose technology that enables further advancements and functionalities that impact today's technology [22]. However, for it to work as intended, it has to be trained. To train ML models, vast datasets are needed and must include relevant information for the data to be usable [15, pp. 1-2]. Hence, metadata tags are used. Metadata tags serve as labels to identify elements within the input for the ML model and are commonly referred to as annotations in the field [15, pp. 1-2]. However, for annotations to efficiently facilitate the learnability of the ML model, they need to be accurate and relevant for the task [15, pp. 1-2].

In practice, when working with AVs, the annotations provide information on elements such as vehicles, pedestrians, roads, and the sky [23]. These annotations are made in both 2D and 3D environments, the latter being significantly more complex since the space includes more dimensions that need to be adjusted [24]. Automation has been integrated into existing annotation systems, enabling a more efficient workflow where annotators can focus on refinements.[24].

2.3 Kognic

Kognic, originally called *Annotell,* was established in 2018 [2]. Kognic specializes in providing an annotation service for ML in the autonomous vehicle sector, helping companies to label, categorize, and track data, to accelerate the ML models for critical tasks. Their mission is aligning data for embodied AI with their software tool, the Kognic Platform, an annotation and visualization tool-set for computer vision.[2]

The Kognic Platform helps enterprises explore, shape, and explain their datasets, see Figure 2.3. It first allows the customers to explore, inspect, adjust, and prepare unannotated datasets in addition to evaluating the proficiency of the model, creating potential strategies to optimize its performance. The next step, shape the dataset, is where Kognic's, or the customer's, annotators mark out and label objects to create their annotations. These annotations are the foundation of the ML models, where the labeled datasets include metadata that is used to teach ML models to work as intended. The final step, explain, analyses how well the performance of the model aligns with the customer's expectations. This could, for example, be through showing the objects that were not included enough in the dataset for the ML model to learn how to handle situations concerning them.[2]

One current optimization problem for Kognic is the shaping phase, with a particular emphasis on enhancing efficiency during annotations. While annotations involve a ”human-in-the-loop” to optimize the accuracy of the results, plenty of automation has been incorporated into the system to streamline the process. Since the process handles vast datasets, small adjustments make a major impact on how much time is spent on each task. There is a fine line between improving the quality through adjustments and performing unnecessary micro-adjustments. This Master’s thesis will investigate the possibility of reducing unnecessary micro-adjustments, potentially having a substantial impact on the overall efficiency of the process.

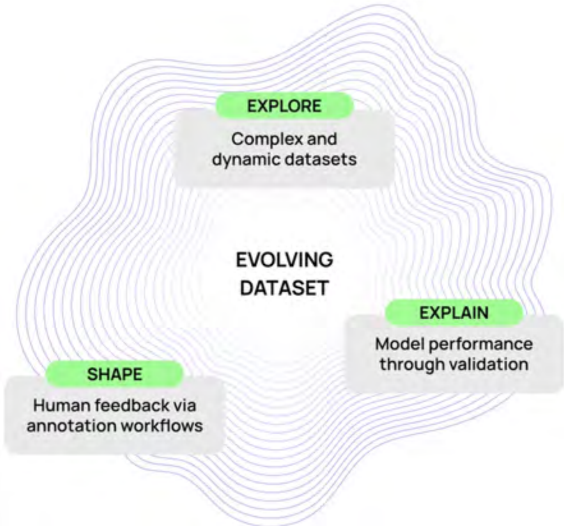


Figure 2.3: Overview of the Kognic Platform. Image provided with consent from Kognic.

3

Theory

In this chapter, theories related or relevant to this Master's thesis will be presented. These could be related to the research question itself, such as motivation to perform well, concepts like wicked problems, and research related to the Kognic Platform, for example, UX in 3D platforms.

3.1 Wicked Problems

Wicked problems are complex problems without a defined solution and with no clear place to begin [25]. The term is now generally well known within the design field and was first introduced by Rittel [26], who describes a wicked problem through ten characteristics:

1. *There is no definitive formulation of a wicked problem.*
2. *Wicked problems have no way of knowing the solution is the final solution.*
3. *Solutions to wicked problems are not true-or-false, but good-or-bad.*
4. *There is no immediate and no ultimate test of a solution to a wicked problem.*
5. *Every solution to a wicked problem is a "one-shot operation"; because there is no opportunity to learn by trial-and-error, every attempt counts significantly.*
6. *Wicked problems do not have a set of potential solutions.*
7. *Every wicked problem is essentially unique.*
8. *Every wicked problem can be considered to be a symptom of another problem.*
9. *The existence of a discrepancy representing a wicked problem can be explained in numerous ways. The choice of explanation determines the nature of the problem's resolution.*
10. *The planner has no right to be wrong.* [26]

The research question of this Master's thesis is a design problem and is in nature therefore wicked with no defined solution. Thus, a *design thinking* approach needs to be adapted to the project, allowing the designer to discover the needs and create a solution for a particular situation. [27]

3.2 Design Thinking

According to Brown in his article *Design Thinking*, [28] design thinking is defined as:

"*...a discipline that uses the designer's sensibility and methods to match people's needs with what is technologically feasible and what a viable business strategy can convert into customer value and market opportunity.*" [28]

Historically, design has been implemented in the final stages of the development process to make the final product nice-looking [28], however, it is now the core of development to accompany user needs. This type of development can be described as having a divergence, transformation, and convergence phase, where you first gather information, then refine and understand the ideas and alternatives, and finally choose between the ideas [29, pp. 64-66]. Another way to explain the phases in design thinking is inspiration, ideation, and implementation [28].

One commonly used process for design thinking is the *Double Diamond* design process which has been further developed into the *Triple Diamond* design process. According to the *Design Sprint Methodology* by *Google* [30], The triple diamond consists of six phases: *Understand*, *Define*, *Sketch*, *Decide*, *Prototype*, and *Validate*, where each diamond represents a divergence and a convergence phase, see Figure 5.1. The first diamond begins with an understand phase where information is gathered and shared between the participants in the project and ends in a converging define phase where the information is evaluated and a focus is set. The second diamond diverges in a sketch phase and converges in a decide phase, beginning with generating ideas and concepts to be evaluated and narrowed down. The final and third diamond starts with a prototype phase where the decided concept is developed to then be tested in the validate phase. Although there are many different ways to explain design thinking and its process, ultimately they all aim to help solve wicked problems. [30]

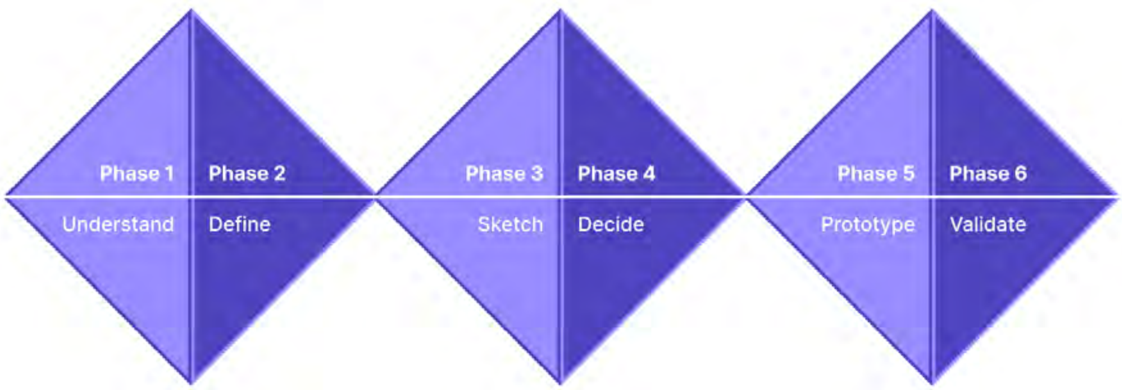


Figure 3.1: The Triple Diamond design process. Figure based on *Design Sprint Methodology* [30]

.

3.3 User-Centered Design

User-centered design (UCD) is a term that originated from Norman and grew popular in 1980 after the book *User-Centered System Design: New Perspective on Human-Computer Interaction* was published [31]. The term sprung from Human-computer interaction (HCI), where HCI focuses on humans' relation to computer products, while UCD is a software design methodology for designers and developers [32, pp. 6]. Another related term is *user experience* (UX), which is used to summarize the whole experience a user has with a software product [32, pp. 6]. The two terms work closely together, where UCD is the methodology used to ensure that software maintains high UX [32, pp. 6]. The UCD method emphasizes the importance of actively involving users in the design process [33]. Since the UCD approach strongly includes the user's perspective, it has proven to be an effective approach to overcome the problems of the traditional system-centered design, where the emphasis was on functionality and the designer built the system on their perception of what is effective [34].

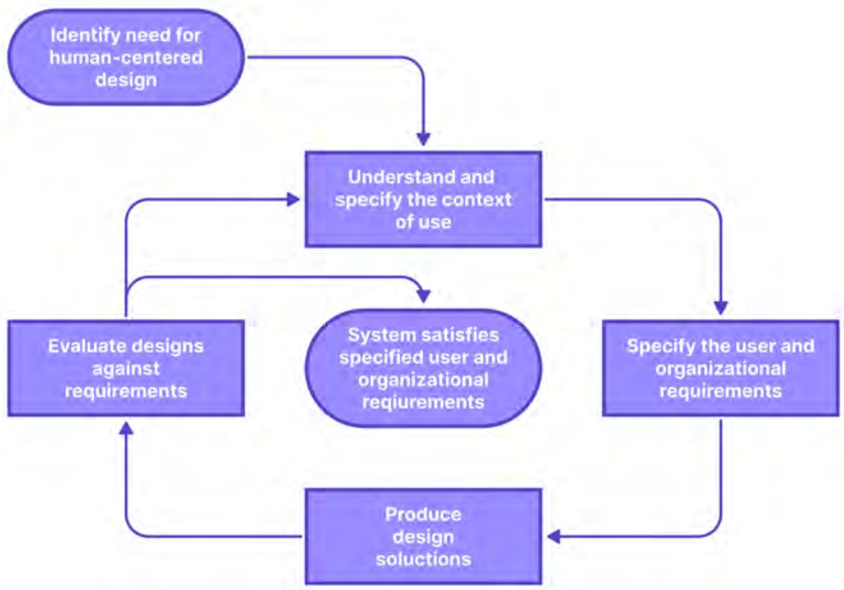


Figure 3.2: Visualization of the user-centered design process. Figure based on Jokela et al. [35].

Upadhyay and Kumbharana [36] identify two key features of the UCD approach. The first is the direct involvement of stakeholders throughout the entire design process and the second is the iterative nature of the method, where the process is carried out in cycles until the usability objectives have been achieved. Additionally, four main activities within UCD methods can be elicited, see Figure 3.2, where Jokela et. al. [35] created a brief description of each activity:

- *Understand and Specify Context of Use:* Know the user, the environment of use, and the tasks that he or she uses the product for.
- *Specify the User and Organizational Requirements:* Determine the success cri-

teria of usability for the product in terms of user tasks, e.g. how quickly a typical user should be able to complete a task with the product. Determine the design guidelines and constraints.

- *Produce Design Solutions:* Incorporate HCI knowledge (of visual design, interaction design, usability) into design solutions.
- *Evaluate Designs against Requirements:* The usability of designs is evaluated against user tasks. [35]

The UCD approach will be beneficial for this Master's thesis since it pertains to a digital design solution that requires efficiency and usability to satisfy the stakeholders. Therefore, integrating users throughout the design process will be crucial to ensure that the solution fulfills the requirements and achieves high usability.

3.4 Task Analysis for HCI

In *The Handbook of Task Analysis for Human-Computer Interaction*, Diaper and Stanton explain why task analysis is the heart of HCI and its use in computer-related projects [37, p. 6]. Any task analysis model should include two aspects:

1. *A description of the domain.*
2. *An explanation of how work is carried out in the described domain.* [37, p. 8]

There are many different ways to create explanations and cognitive user models for the second aspect of task analysis, for example, *hierarchical task analysis*, *Venn diagrams*, *KLM-GOMS*, and *flowcharts*. However, the goal of performing a task analysis is to improve computer systems and increase the usability for users [37, p. 20]. As mentioned, task analysis is an essential part of HCI activities. The behavior of people is much more complex compared to the information processing of a computer [37, p. 21] and needs to therefore be in focus. This is a foundational problem for task analysis and thus, a cognitive user model is essential [37, p. 21].

Since this Master's thesis falls within UCD, a theoretical framework with many connections to HCI, task analysis theories are beneficial to fully understand how the user interacts with the system.

3.5 Design for Sovereign Applications

Cooper et al. [38] explain that sovereign posture applications are applications used for long continuous periods of time by their users. These applications contain a large number of functions and features and are built to be maximized, taking up the entire screen. When designing a sovereign application, one must consider users with all levels of experience. Beginners should rapidly and painlessly be able to become intermediate users, intermediate users should not be blocked from becoming experts, and expert users should be provided with the tools they require. Additionally, one should be generous with screen real estate, use a minimal visual style, provide rich visual feedback, and support rich input. [38, pp. 208-211]

Since the Kognic Platform is a sovereign application, these aspects need to be considered during this Master's thesis. Additionally, most users are experts in using the platform which also needs to be taken into account. Expert users look for esoteric features that could be useful for them and require quick access to their commonly used tools, experts want shortcuts to everything [38, pp. 243-247].

3.6 UX in 3D Platforms

UX is a widely researched topic in platform design, with the primary focus being on basic desktop applications, neglecting the fact that the field has matured and the complexity of systems has grown [39]. People have created frameworks to propose important aspects of what makes a system user-friendly. For example, Molich and Nielsen [40] proposed 10 general rules to generate a heuristic user interface design. Due to the growth of today's systems, especially when considering 3D environments, these models have been proven to be insufficient [39]. When working with complex systems, previous UX and *user interface* (UI) guidelines need to be developed. Lee et al. [39] proposed a framework applicable to complex 3D design and engineering systems, including general UI principles that are relevant for more complex systems, principles specific to 3D parametric design, and principles for user support. Bellow, the following principles are cited from *Usability principles and best practices for the user interface design of complex 3D architectural design and engineering tools* by Lee et al. [39]:

Principles for general system design:

- *Viability: Making relevant information conspicuous and easily detected.*
- *Feedback.*
- *Consistency: Uniformity of system semantics across similar situations.*
- *Recoverability: Providing the users with options to recognize and recover from errors.*

Principles specific to 3D parametric design:

- *Maximization of Workspace: Providing maximum screen space.*
- *Graphical richness: Replacing textual information with graphical information.*
- *Direct Manipulation: Providing interactions that are perceived as direct.*

Principles for user support:

- *Familiarity: Leveraging user's knowledge and experience in other real-world or computer-based domains.*
- *Customizability: Support to explicitly modify the interface or operability of the system based on the user's preference.*

- *Assistance: Providing support both explicitly and implicitly.*
- *Minimalistic design: Keeping it simple to minimize redundancy of information.*
- *Context recognition: Automatic adjustments of the interface or operability of the system depending on user mode.* [39]

Most of the principles listed are directly applicable to the final design solution of this Master's thesis. Nevertheless, it is crucial to constantly keep all principles in mind when introducing new functionalities into a complex 3D system. Since this project considers a complex system, all functions are somewhat interconnected and alterations in one area may affect other parts. Failure to consider the principles during the project might compromise the usability, potentially doing more harm than good.

3.7 Gamification in Annotation

Gamification is a concept utilizing game mechanics to make an assignment more interesting, often including elements of reward to enhance motivation [41, p. 4]. According to Marczewski, gamification can be defined as:

"*The application of gaming metaphors to real life tasks to influence behavior, improve motivation and enhance engagement.*" [41, p. 4]

In a study conducted by Mekler et al. [42], the impact of integrating gamification in an annotation tool was investigated. In their research, users' behaviors when annotating images were tracked for different gamification mechanisms to assess if they could enhance their intrinsic motivation. The elements tested during the study were points, leaderboards, and levels, comparing them to a plain condition. The findings revealed that incorporating gamification did not significantly affect the users' intrinsic motivation, however, it did affect the number of annotated tasks without compromising the quality. [42]

Since this Master's thesis concerns an annotation tool, it is interesting to explore how gamification can be utilized when designing to minimize micro-adjustments. Especially since previous research has proven that it can increase the quantity without affecting the quality of the annotations, which suggests that incorporating gamification elements can be a suitable solution.

3.8 Motivation and Perfectionism

Motivation is a person's incentive that gives meaning or guidance to behavior, whether consciously or unconsciously [43]. Motivation can be divided into two different categories, *intrinsic motivation*, a person's internal incentive to act, and *extrinsic motivation*, driven by external factors [44]. Generally, when speaking about motivation, the incentive of performing a task is a combination of them both [45, pp. 7-10]. For example, in a work environment, external incentives exist in the form of salary and deadlines [46], while intrinsic motivation exists simultaneously in the form of job satisfaction and the achievement of a goal set by the worker [45, pp. 7-10].

In a work environment, most employees need motivation to feel engaged and perform well [47]. One source of motivation for employees is accomplishments and achievements for reaching personal or professional goals [47]. Unmotivated employees tend to spend insufficient time on a task and produce low-quality work [47]. However, there is the other side of the spectrum too, *perfectionism*. Perfectionism can be defined as aiming for excessively high standards [48] and can result in negative outcomes, one being a decrease in efficiency [49]. It can be a hard task for an individual to set the boundaries of what is good enough. One solution, cited from Ashford [50], to solve this problem is a self-assessment process, including three tasks:

1. *Establish the specific standards on which they should judge their performance.*
2. *Learn the feedback cues to which they should attend among the many available cues.*
3. *Interpret those cues.* [50]

The problem of micro-adjustments could be related to perfectionism since they are a product of execution that exceeds expectations and needs. Understanding the concept is valuable to this project, enabling a broader understanding of the underlying cause of micro-adjustments. Especially the self-assessment task concerning feedback, since feedback can be provided in a user interface.

Since intrinsic motivation within a work environment includes job satisfaction, which is connected to how well the users feel like they perform, it is important to be acknowledged for the design in this Master's thesis. To maintain highly motivated users, the redesign needs to provide a solution that streamlines the process while maintaining the user's satisfaction.

3.9 Nudging

Nudging involves any aspect of design that predictably alters user behavior, without preventing options or making major changes to economic incentives [51]. To understand how nudging works, it is important to have a brief understanding of how humans make decisions. Human decision-making can be divided into two systems, system 1 and system 2 [52, pp. 16-22]. System 1 is responsible for the automatic principle of thinking, which is quick with little to no effort [52, pp. 16-22]. System 2 handles the reflective part which involves higher complexity tasks, often associated with concentration, choice, and subjectivity [52, pp. 16-22]. The systems work cooperatively, where system 2 engages when system 1 is not sufficient for the task as it is human nature to minimize effort [52, pp. 16-22]. Since the human brain works accordingly, humans tend to take the easiest path when executing a task [51]. This is when nudging comes into play. Nudging utilizes the automatic decision-making process, altering the user's behavior by influencing the decision-making in predictable ways [51].

A framework of how to incorporate nudging in HCI was created by Caraban et. al. [51], clustering 23 different aspects into six general categories: *Facilitate, Confront, Deceive, Social Influence, Fear,* and *Reinforce*. The aspects from this framework deemed to have the most potential in this project are; *deceptive visualization* from the category deceive, and *ambient feedback* and *subliminal priming* from the category reinforce. Deceptive visualization leverages that people are more focused on items and information that is more prominent, using optical illusions to alter people's perception and judgment. Ambient feedback instead reinforces users' behaviors with small changes to not disrupt their current activity. Subliminal priming increases preferences for certain stimuli by notifying them subconsciously. [51]

These aspects could be of potential use during this Master's thesis as they can nudge users to change their behavior subconsciously, which could be a possible strategy to minimize micro-adjustments without interfering with the overall workflow.

4

Methodology

This chapter consists of the various practical methods used and how they are relevant to this Master's thesis. The project will follow a design thinking approach, see Section 3.2, with the triple diamond design process as its basis.

4.1 Literature Review

The basis for all academic research is building your research on and connecting it to existing knowledge. This task has become increasingly more complex since it is constantly evolving, making it difficult to keep up with state-of-the-art research. Because of this, utilizing *literature review* as a research method is especially important in today's academia. The literature review should include collecting and synthesizing previous research, which sets the foundation for creating a theoretical framework. [53]

This project will follow an integrative research review, which aims to assess and evaluate literature relevant to the topic to enable new knowledge to emerge.

Gall et al. in *Education research: An introduction* [54], highlights some of the common mistakes researchers usually make, some of which are mentioned below:

- *The researcher does not relate the findings to their research.*
- *The researcher does not take sufficient time to identify the best sources to use.*
- *The researcher uses secondary sources rather than primary sources.*
- *The researcher does not state the search procedures used.*
- *The researcher does not consider or ignore contrary findings and different interpretations.* [54, pp. 161-162]

These aspects will be crucial to refer back to while conducting the literature review of this Master's thesis during the understand divergence phase of design thinking, see Section 3.2.

4.2 Understanding the Domain

Before getting in contact with users, it is essential to understand the domain that is being designed for. This is because the stakeholders the researcher will approach throughout the user research phase will expect basic knowledge. Hence it is important to recognize commonly used terms and understand the process to talk about the product in an explorational way. Baxter et al. [55, pp. 25-35] have formalized the process one should follow to gain thorough knowledge within the domain before conducting the user research, below the steps that are deemed applicable to this Master's thesis are listed:

- *Use your Product* - The researcher should interact with the product, push its limits, and see how it responds. Through this, potential pain points can be distinguished, which makes it easier to spot and discuss them when the user encounters them.
- *Networking* - Talk to knowledgeable people in the company that produces the product. By consulting them, knowledge about the artifact and previous findings can be collected.
- *Log Files and Web Analytics* - By analyzing log files, one can usually gain valuable insights into how the users interact with the product. Log files often include a path of which actions were taken by individuals, where the user clicks, and how much time is spent on certain parts of the product.
- *Competitors* - Through an analysis of existing competitors' products, including for instance what features they use, their strengths, and their weaknesses, an effective way to gain advantage over competitors can be created. A competitors' analysis should not be limited to only direct competitors, it should include surrogate products as well. Surrogate products are products with similar features, which can give valuable knowledge of what could be incorporated and what does not work.[55, pp. 25-35]

Since this Master's thesis involves a complex system, it is essential to understand the domain before approaching stakeholders. Collecting valuable information regarding how the platforms work will enable further understanding of what areas to investigate further and allow for a good basis for future work.

4.3 Observations

Observation is an ethnographic method designed to identify unspoken areas for potential insights, offering a nuanced understanding of a user's interaction with a product [56, p. 85]. During observations, users are observed to identify how they interact with a product, and data is collected to understand their behavior [57].

There are different types of observations, the two most common forms can be divided into *controlled observation* and *naturalistic observations.* Controlled observations are usually conducted in a laboratory setting where the users know the purpose of the observation and why they are being observed. The advantages of this type of observation are that it is easy to reproduce, easy to analyze, and quick to conduct. However, since the users know that they are observed, they tend to change their behavior which can affect the results. Naturalistic observations are instead conducted in the user's natural settings, in a less structured manner. The advantages of naturalistic observations are that they generate more reliable results since the observations will record the real-life usage of the product. However, they tend to be hard to replicate and external factors can affect the results. [57]

Nowadays, video-based observation methods are commonly used when studying how users interact with technology [58], one of them being screen-capturing. According to Bao et al. [59] who conducted a study where they analyzed developers interacting with their software tools, screen-captures can generate valuable insights into how software is being used and how it can be improved. Even if the user group in this project is not developers, the users work within a highly complex platform, interacting with various tools. Consequently, the resemblance suggests that screen-captures serves as an effective tool to capture and investigate the user's workflow in this project. A screen-capture observation falls under the category of natural observation since the users will be focused on their regular tasks in their natural settings. During this project, screen-captures will be used during the understand divergence phase of design thinking, see Section 3.2.

4.4 Flowcharts

Flowcharts is a method used to visually represent user flows and tasks [60]. They are especially useful for explaining and showing interactions in the design as they are comprehensible and easy to understand [60]. Symbols are utilized to represent different actions, squares represent a step in the process, arrows represent the direction the user moves in, and diamonds represent a point where decisions need to be taken, see Figure 4.1 [60]. Flowcharts will be used during the understand divergence phase of design thinking, see Section 3.2, to get a comprehensive understanding of the process of annotations.

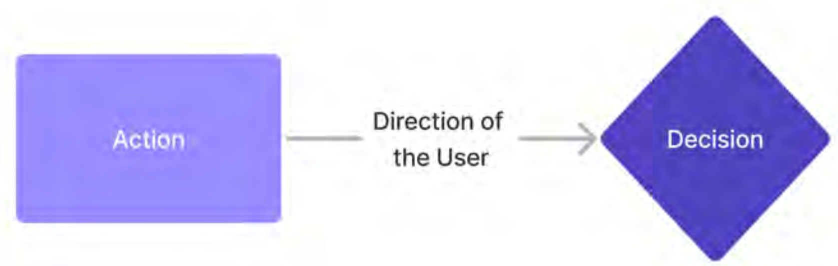


Figure 4.1: The symbols used for representing different actions while creating flowcharts.

4.5 KLM-GOMS

The *GOMS* concept is useful for analyzing how to complete a task since it decomposes it into goals, operators, methods, and selection rules. Even if the concept itself is quite general, it has generated a family of extensions that collect quantitative and qualitative data on people's usage of a system. [61]

The *keystroke level model* (KLM) is one of the methods in the GOMS family. It allows for a decomposition of the general tasks into unit tasks, such as mouse strokes and clicks, which in turn allows for predicting human performance within a system [62]. The unit tasks have predetermined times it takes for the user to complete, for example, moving the hand to the keyboard from the mouse takes 0.40 seconds [63]. The method generates the time it takes for a user to complete a task by adding up all unit tasks [62].

The KLM-GOMS method proposes advantages by predicting performance, however, it presents some limitations as well. For example, the method assumes that the users are experts, does not account for any human errors, and only considers linear routine tasks, neglecting the fact that users reach a goal in different ways [62]. Additionally, deciding how many mental acts of thinking should be included and where they appear in the task is the most difficult part of the KLM-GOMS method [63].

Despite the disadvantages, the KLM-GOMS method can be considered a valuable asset to this Master's thesis and will be used during the understand divergence phase of design thinking, see Section 3.2. Since the method will result in the optimal time to complete a task in the platform, it allows the possibility of comparing the optimal to the actual time, possibly generating an approximation of the time loss caused by micro-adjustments.

4.6 Semi-structured Interview

Semi-structured interviews are a qualitative research method with high flexibility to adapt the format of the interview depending on the participant's responses. In the article *What are Semi-Structured Interviews?* by the *Interaction Design Foundation*, they list three key characteristics of semi-structured interviews: [64]

- *The flexible nature allows researchers to dive deeper into the topic and adapt the interview based on new insights.*
- *Their emphasis is on participant perspective and experience.*
- *They are often used in research projects that aim to generate new ideas or theories rather than test existing ones.* [64]

Since semi-structured interviews are flexible, all questions are not set beforehand, instead, they consist of a combination of different question types to enable exploration. The questions that are commonly used can be categorized into three main areas: [64]

- *Open-ended questions* - The characterization of these questions is their broadness, allowing the user to unrestrictedly present their thoughts and experiences.
- *Closed-ended questions* - These are more specific and often provide predetermined answers, such as yes and no questions. These types of questions allow for, for example, collecting data about specific attitudes or behaviors.
- *Probing questions* - These are follow-up questions aimed to elaborate and clarify topics, used to dig deeper into the participant's responses.[64]

When conducting a semi-structured interview, it is required to have some previous knowledge about the area that is being researched [65], since the interview questions are based on that knowledge, laying the foundation for what topics are to be investigated [66]. Thorough pre-studies will therefore excel the results from the interview since the interviewer will possess useful knowledge to understand what questions to ask, and what areas could be interesting to investigate further. Semi-structured interviews are a valuable resource for this project, to gain knowledge about the users' experience and insights into what needs to be considered when designing for them. This method will therefore be used during the understand divergence phase of design thinking, see Section 3.2.

4.7 UX Workshop

According to Kaplan [67], *UX workshops* are defined as: "*Intensive collaborative sessions used to solve problems and enable progress on a particular challenge throughout the design timeline.*" She describes two steps that should be used when planning a user research workshop:

1. Invite core team members, key stakeholders, and users, if possible and appropriate. Anyone with key knowledge of user research is a good candidate for an empathy-workshop participant.
2. Create an informal environment that encourages brainstorming from a user's perspective. [67]

One key distinction between traditional meetings and workshops is their structure, with the workshop encouraging participation and activity, while traditional meetings tend to be passive, where participants are meant to learn and listen. Another aspect of differentiation is the purpose, as workshops are conducted to solve a problem, develop a plan, or reach a decision, rather than sharing information. [67]

UX workshops are a suitable approach during the design process, enabling opportunities where a group of participants together can work on achieving a certain goal. Through the integration of people, new perspectives and considerations may surface, enabling a more comprehensive understanding that might otherwise be overlooked.

When talking about UX workshops as a research methodology, it is an area that has yet not been extensively explored [68]. However, Ørngreen [68] argues that workshops for research have a unique ability to inspire new insights, and in combination with other empirical approaches, can be a valuable resource during the exploratory phase. This suggests that conducting a UX research workshop, in conjunction with other methods, during the understand divergence phase of design thinking, see Section 3.2, can yield useful and valuable insights into this project. In this project, a UX research workshop could be conducted with knowledgeable people at Kognic who have previous experience with micro-adjustments.

4.8 Affinity Diagramming

Affinity diagramming, also referred to as the *KJ method*, is a method used to sort unstructured qualitative data, such as ideas, information, and observations to discover patterns that can generate insights [69, p. 22], [70]. The gathered data is analyzed and clustered based on similarity into manageable groups, which later can be used to draw informed conclusions [70]. Lucero [71] has improved the method for industrial and academic purposes to be more suitable for small to medium-interaction design projects and divided the process into four steps:

1. *Creating Notes* - Gathering all valuable insights, such as quotes from interviews, notes from observations, and other issues that have arisen from the user research on sticky notes.
2. *Clustering Notes* - The notes are spread out, read, and clustered into groups.
3. *Walking the Wall* - The clustering of notes should be done in an iterative process, allowing for changes, for example, if any general topics are missing and if any clusters can be combined, supporting new connections.
4. *Documentation* - The created connections and group topics are written down and shared with relevant team members. [71]

Affinity diagramming is a valuable asset during the define convergence phase of design thinking, see Section 3.2, for this Master's thesis to handle the collected data in a structured way, allowing for identifying patterns and potential contradictions.

4.9 Requirement Specification

To define the user needs, a specification of design requirements and guidelines is necessary to summarize the functionalities, capabilities, and characteristics of the final design solution [72]. The requirements and guidelines act as a basis for the remainder of the design process. This is done to ensure the user needs are met, thus its importance to this Master's thesis and a part of the define convergence phase in design thinking, see Section 3.2.

The process begins with eliciting valuable insights from the user research results deemed as important to acknowledge for the final design solution to work. To prioritize and organize the insights, the *MoSCoW* method can be utilized. When conducting the MoSCoW method, the insights are divided into four categories [73]; *Must have, Should have, Could have,* and *Won't have,* creating a structure of what is vital, important, useful, and insignificant to consider. Must have are requirements, should have and could have are optional but desirable features, and won't have are features that will not be incorporated due to, for instance, being too expensive to implement. [73]

4.10 Ideation

The ideation process is a powerful activity that aims to produce a large number of ideas on a certain topic, always without an attempt to assess their likelihood of success or reasonability [74]. The ideation process is utilized for innovation, where ideas can take multiple different forms, such as verbal, visual, concrete, or abstract [75]. There are many different methods of ideation and the ones used in this project are explained below. Important to note is that all ideation methods serve the same purpose, generating quantity instead of quality [74].

4.10.1 Brainstorming

Brainstorming is the process of generating ideas that solve a previously defined design problem [76]. There should be some rules and limitations to keep the brainstorming session on track, but out-of-the-box and lateral thinking is essential [76]. The concept of brainstorming was first introduced by Osborn in his book *Applied Imagination: Principles and Procedures of Creative Thinking* [77]. This book introduces some rules to the brainstorming process which have been further developed into what it is today. According to the *Interaction Design Foundation* [76], there are eight general rules to be followed when brainstorming:

1. *Set a time limit* - 15-60 minutes is the standard.
2. *Begin with a target problem and brief* - Everyone should be on the same page and stay on topic.
3. *Refrain from judgment and criticism* - No idea is a bad idea.
4. *Encourage weird and wacky ideas* - It is easier to tone down an idea than to come up with a new one.
5. *Aim for quantity* - Quantity generates quality.
6. *Build on others' ideas* - Allowing other's ideas to trigger their own and reach new insights.
7. *Stay visual* - Post-Its or other visualization tools.
8. *Allow one conversation at a time* - Helps keep the session on track and respect each other. [76]

The brainstorming process will be essential in this Master's thesis to generate ideas and come closer to a solution to the research question. It will be the first step in the sketch divergence phase of design thinking, see Section 3.2, and will act as a good basis to keep the ideas and solutions flowing.

4.10.2 Sketching

Sketching is a fast, easy, and cheap method to create new ideas or develop and iterate on previously generated proposals [78], [69, p. 116]. It is one of the clearest ways to communicate ideas and helps eliminate any misunderstandings [78]. Sketching can be done alone or in collaboration, called *Design Charrettes*, with the benefits of being inspired by others, helping designers who are temporarily out of ideas, and hearing the priorities of others [79]. In this project, sketching will be used to explore and propose ideas during the sketch divergence phase of design thinking, see Section 3.2.

4.10.3 Mind Mapping

Mind mapping is a method utilized to create representations of aspects concerning a specific theme, visualizing how they are related to each other. It can be used for various purposes, one being to explore the solutions space for a defined problem area. The method is conducted by writing the main theme in the middle of a large piece of paper or a whiteboard. Afterwards, a brainstorming session about aspects concerning the main theme is carried out, where connections depending on how they relate to each other are created [56, p. 45].

In this project, mind mapping will be utilized to explore potential solutions during the sketch and divergence phase of design thinking, see Section 3.2

4.11 Prototyping

Prototyping differs from sketching in that it aims to test specific features and general ideas [78]. The prototypes can be either low-fidelity (lo-fi) or high-fidelity (hi-fi), each with its benefits and disadvantages [80]. However, they both let designers test their ideas before implementation and production, allowing for iteration if it is not up to the right standards [80].

4.11.1 Low-fidelity Prototyping

Lo-fi prototyping, or *Wireframing*, is the process of drawing digital interactive products to establish the fundamental structure and a flow of the interactions for various design solutions [81], [69, p. 136]. Wireframing can be done either on paper or digitally, however, it must be nondetailed and focused on the structure of the interface using block design [81]. This leads to figuring out the hierarchy of the interface and the priority of its various elements [81]. Wireframing will be utilized during the sketch divergence phase of design thinking, see Section 3.2, to further explore the ideas and potential solutions from the ideation process.

4.11.2 High-fidelity Prototyping

Hi-fi prototyping is the process of creating a finished simulation of the final design solution, which gives a realistic look and feel to the product [82]. Additionally, the hi-fi prototype can accommodate interaction within the application which can be used as a demonstration for stakeholders or user testing [82]. In this Master's thesis, hi-fi prototyping will be used for this exact purpose during the prototype divergence phase of design thinking, see Section 3.2. To design the prototypes, the application *Figma* [83], an industry-leading user-friendly design and prototype tool, will be utilized.

4.12 Evaluation

There are various methods for conducting user experience evaluations, such as user interviews, user testing, heuristic evaluation, etc [84]. However, they all aim to help understand what works, what does not, and what improvements can be made [84]. The evaluation methods used in this Master's thesis and why they were chosen are described below.

4.12.1 A/B Testing

According to the Interaction Design Foundation [85], *A/B testing* is used to identify performance by having users test two different versions of the same design. The A design is the original version while the B design is the new version. It is used to test minor changes such as individual elements or minor layout changes while everything else in the design stays the same. It is especially efficient since it is a quick, easy, and cheap method to test these minor changes, clearly showing which design is superior. [85]

A statistic analysis is then conducted to compare the different versions, for example, the difference in the number of clicks on specific aspects such as if an item has been added to the cart or how many of the users signed up for a newspaper [85]. A/B testing will be important for this Master's thesis during the validate convergence phase of design thinking, see Section 3.2 since it allows for accurate statistical analysis of the current design compared to the new design.

4.12.2 Cognitive Walkthrough

According to Lewis and Wharton [86], *cognitive walkthrough* is a method used to evaluate user interfaces from a cognitive perspective, to identify problems within the interface, and to suggest areas for improvement. The method is carried out by an analyst who chooses a specific task to evaluate from a cognitive perspective. Following, the analyst creates one or more paths of actions in which this task can be completed, whereas each action is assessed by the analyst, deciding if a hypothetical user will choose the right action or not. The method is useful to evaluate the usability of an interface in a quick and cost-efficient way and allows for evaluation without a functioning prototype or involvement of users. [86]

According to the Interaction Design Foundation [87], a cognitive walkthrough begins with deciding which tasks are to be evaluated, these tasks are broken down into smaller actions, allowing the analyst to assess each step of the procedure to finalize the task [87]. To assess each action, Blackmon et al. [88] created four assessment questions which are as follows:

- *Will the user try and achieve the right outcome?*
- *Will the user notice that the correct action is available?*
- *Will the user associate the correct action with the action they expect to achieve?*
- *If the correct action is performed, will the user see that progress is being made towards their intended outcome?* [88]

Through these questions areas of improvement can be identified, as well as when during the task these actions occur. This method will be utilized during the Master's thesis during the decide divergence phase of design thinking, see Section 3.2 to be able to evaluate the concepts without user involvement.

4.12.3 User Experience Surveys

There are many different types of *user experience surveys*, all of which have various types of purposes for different situations [84]. Closed-ended questionnaires are used to gather quantitative data to decide which design concepts are preferred [84], this method will be utilized in the Master's thesis during the decide convergence phase of design thinking, see Section 3.2. Writing good surveys with unbiased questions is not easy, but best practice guidelines exist such as asking about the right things, using clear language, asking one question at a time, and the use of balancing scales [89].

5 Planning

This chapter presents the planning of the Master's thesis. It includes the time plan and planned result.

5.1 Time Plan

The thesis will be carried out for 20 weeks with a start date of January 17th and include the following main areas of work: planning report, user research, ideation, prototype & evaluation, and thesis report. Throughout the project, consultations with both the supervisor from Chalmers University of Technology and the advisors from Kognic will be held weekly. For the complete time plan with all sub-sections, see Figure 5.1.

5.2 Planned Result

The planned result is to answer the research question and provide new knowledge by utilizing the design process and interaction design methods, such as literature research and analysis, requirement elicitation, and concept development. The goal is to reduce micro-adjustments by increasing annotator efficiency through design. The results will include design guidelines on how to reduce micro-adjustments during annotation work and a final design solution.

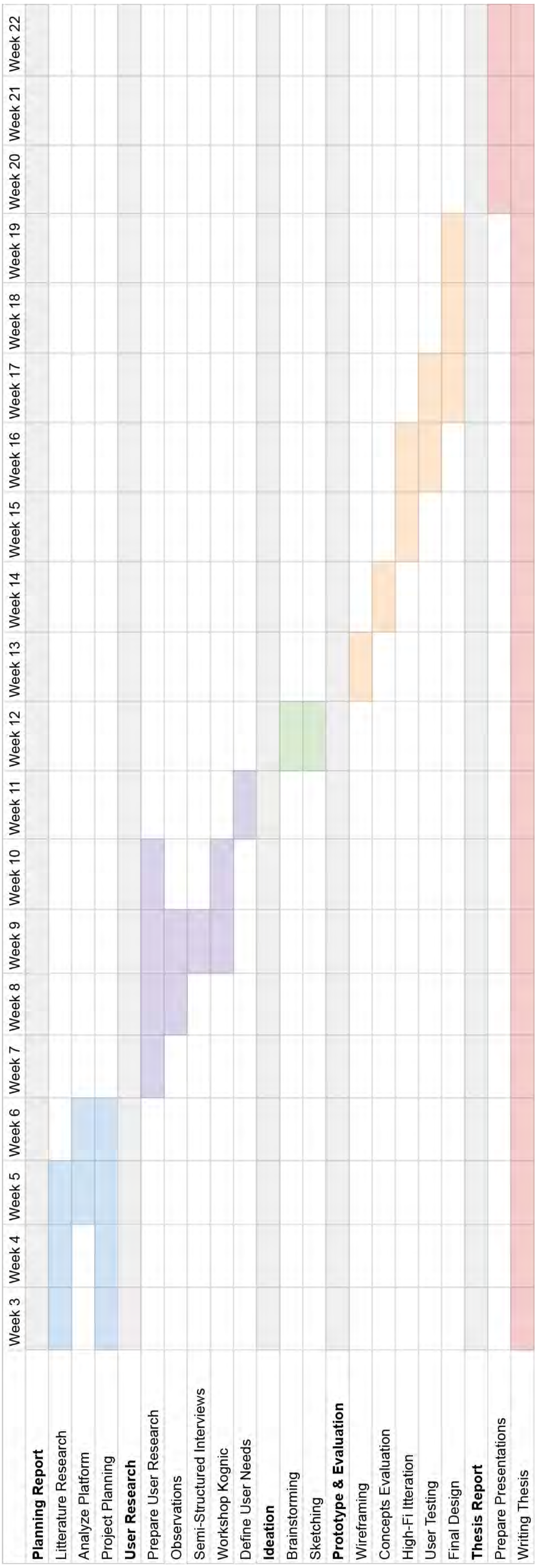


Figure 5.1: Complete time plan for the Master's thesis.

6

Ethical Issues

This chapter will discuss any potential ethical issues that may arise during the Master's thesis, including the adherence to fair working conditions and considerations related to the storage of personal information.

6.1 Business Process Outsourcing

AI is expanding significantly in today's society, yet it is important to remember that AI does not simply appear out of thin air. At the core of functional AI services are labeled datasets, serving as the foundation for the ML models. Even if plenty of automation has been incorporated into the annotation system to streamline the process, the final quality still depends on reliable human refinements. Now there are firms with the sole purpose of providing annotations, these firms are referred to as *Business Process Outsourcing (BPO)*. Recently, concerns regarding the work conditions at the BPOs have arisen. This is something that Kognic acknowledges and actively opposes by setting up regulations for their subcontractors to ensure fair labor conditions. [90]

Kognic has established a code of conduct, where they provide a legal agreement regarding regulations applicable to all subcontractors in their supply chain. The code of conduct contains, among other things, regulations regarding annotator worker's rights, unionization, work time, salaries, and personal information. [91]

Additionally to their code of conduct, Kognic has written an article named *Fair Working Conditions & Global Dependencies in Machine Learning* [90], where they further elaborate on their process of creating fair work conditions. Some actions they take to ensure this are to receive anonymous feedback from annotators through surveys, continually work on developing the platform to make it user-friendly, minimize the effort of manual annotation, occasionally visit the BPOs for an accurate picture of the work environment and condition, and careful selection of BPOs, where companies that value profit over people are sorted out. [90]

This Master's thesis can, through the reduction of micro-adjustments, aid in providing the right tools and processes for annotators. If successful, it could support better work conditions since it will improve the platform's user-friendliness, reducing the manual effort needed during the annotation process.

6.2 Storage of Personal Information

Since the annotators are based outside of Sweden, methods need to be adapted to observe the annotators, interview them, and evaluate the final design solution. These phases will require more unconventional procedures than what would usually be the norm, such as observations through video recordings, digital interviews, and unmoderated remote usability testing. Naturally, the storage of personal information is a concern since the details gathered need to be stored during the evaluation and analysis of the data.

This needs to be considered during the project and will therefore follow the regulation (EU) 2016/679, otherwise known as the *General Data Protection Regulation* (GDPR) [92]. This regulation is also applicable to Kognic and the personal data handled by Kognic [91], which includes the data gathered during this Master's thesis. Kognic also states in their *Code of Conduct*,

"*Kognic and its suppliers shall when entrusted with personal information, safeguard it and take appropriate steps to protect it from misuse. Kognic and its suppliers observe all applicable privacy laws when we collect, use, and share personal information about individuals.*" [91]

This will also apply to this project and will therefore comply with Kognic's policies. Personal information that will be handled in this project are screen and audio recordings, no information regarding name, address, gender, or age will be collected. Everything will be stored in Kognic's *Google Workspace* which data region is configured for Europe in compliance with GDPR.

7

Process and Execution

In this chapter, the execution of the design process and the various interaction design methods utilized are explained. It will show the results of the phases in design thinking, see Section 3.2. The first section will contain the understand and define phases, the second section will concern the sketch and decide phases, and the third section will be the prototype and validate phases. This chapter is the core of the Master's thesis and will output a final design solution.

7.1 Phase 1: Understand and Define

This section, see Figure 7.1, concerns the exploratory phase of design thinking, explaining the process of how the data was collected and analyzed. The output of this section will stand as the foundation for the design, an understanding of the problem area, and a requirements specification.

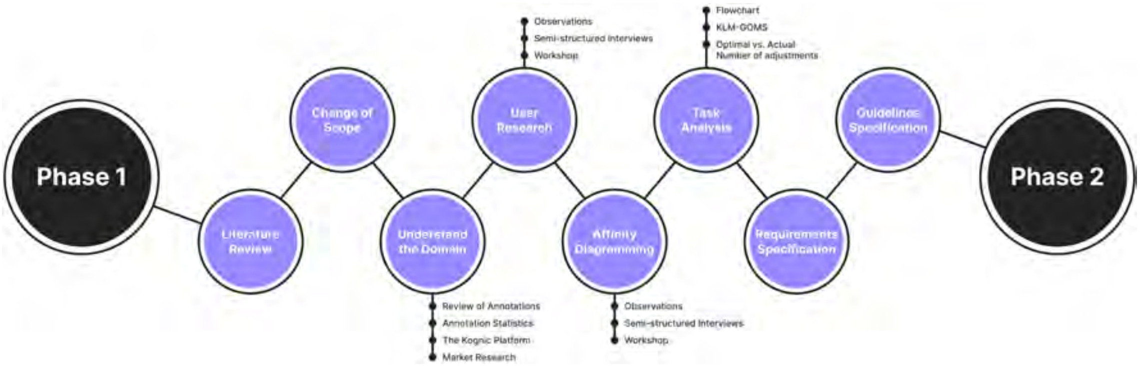


Figure 7.1: Phase 1: Understand and Define.

7.1.1 Literature Review

To generate an understanding of the AVs sector and related fields that could be of importance to this Master's thesis, a literature review was conducted, see Sections 2, 3, and 4. The literature review involved a deep dive into AVs, including how they function, the hardware and software beyond them, how the software is trained to work as intended, and their potential impact on our society. Additionally, literature about methodologies, mainly from *Nielsen Norman Group* [93] and the *Interaction*

Design Foundation [94], were gathered and foundational theories of interaction design were collected. *Google Scholar* and *Chalmers Library* were the main platforms for finding sources for the literature review. Searches include keywords such as ”autonomous vehicles”, ”prediction models for autonomous vehicles”, ”autonomous intelligence”, ”annotation for machine learning”, ”UX in 3D platforms”, ”design thinking”, ”user-centered design”, ”motivation”, and ”perfectionism” were used.

Through the literature review, knowledge regarding the AVs sector, theories, and methodologies were established. However, since annotation in 3D point clouds is still quite a novel field, there were few articles exploring the design within the platform used for these tasks. Instead, articles about general design principles, for example, design thinking, user-centered design, and UX in 3D platforms were collected, see Sections 3.2, 3.3, and 3.6 respectively. These helped with a broader understanding of what could potentially be of importance when designing for 3D point clouds.

The literature review's findings are outlined in Chapters 2, 4 and 3.

7.1.2 Change of Scope

Initially, the focus of the project was investigating single-frame point cloud annotations of cuboids. However, after consulting our advisors at Kognic, a suggestion about changing the scope to sequences was brought up. This was suggested since sequences could potentially give more insights into micro-adjustments. The meeting resulted in a change of scope, with the new focus including both single-frames and sequences. Since there are similarities during the annotation of these two types of tasks, it would not be difficult to adapt and broaden the scope. Rather, it would assist in the work of this Master's thesis, since more information could be gathered and utilized during the project.

7.1.3 Understand the Domain

To better understand the domain, exploring and researching competitors and the product is crucial, following the process outlined in Section 4.2. This section presents the result of the market research, a deep dive into the Kognic Platform, the process of reviewing annotations, and statistics of annotation.

Market Research

Researching the existing market was carried out by consulting the advisors at Kognic about their competitors and analyzing their web pages to gather information. Additionally, companies working with annotation other than the ones provided by Kognic were explored.

There are many different companies within the annotation service for the ML models sector. They focus on different use cases, ranging from text, to image, to video annotations, with some providing all of them. Additionally, annotation services are available for various industries, including automotive, sports, healthcare, agritech, retail, security, and surveillance. Again, some companies focus on one industry while others try to provide their service for all of them.

The problem with researching the market is that the established annotation tools are not available for public use, and if they are, only a limited test version is accessible. Some provide promotional images and videos on their website, but this has little to no use. Since the problem of micro-adjustments is a very detailed and specific area, the test versions and visual media found can not be utilized to research, find inspiration, or discover potential solutions to this topic. Therefore, the result of the market research is that there are many competitors in the field, however, it did not help with specific platform-related details and solutions.

The Kognic Platform

This section will go more in-depth into how annotators work with marking out objects and the tools they use. The knowledge concerning the platform presented below was gathered through interacting and using the platform. This was combined with information gathered from the observations and the semi-structured interviews, which can be read in Sections 7.1.4 and 7.1.4. Figure 7.2, shows the workspace used by annotators within the Kognic Platform. On the left-hand side are the tools used during annotation, the right-hand side is an index of the various object classes, and the bottom displays camera views, the frames within the dataset sequence, and its controls. The annotator can also access the task guideline through the help menu in the menu bar. This gives all the instructions the annotator needs to give the customer the desired dataset, including, but not limited to, what classes should be used and how accurate the annotations should be.

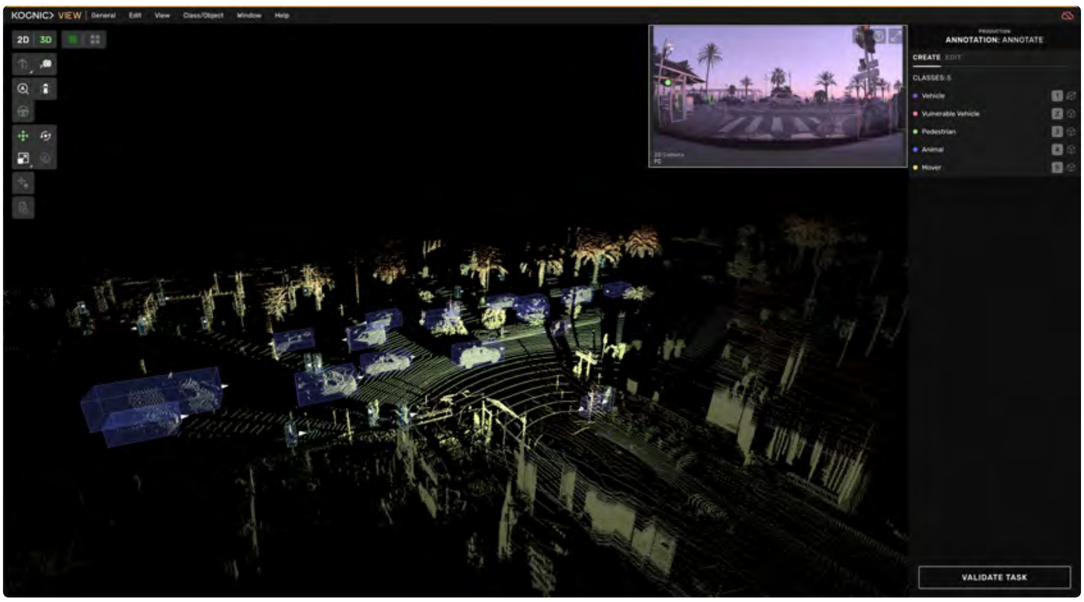


Figure 7.2: An overview of the 3D view while annotating in the Kongic Platform. The vehicles are marked out using cuboids. Screenshot provided with consent from Kognic.

Figure 7.3: The tools used for creating annotations and working the Kognic Platform. Screenshot provided with consent from Kognic.

Tools In the Kognic Platform, there are various tools available to help create annotations which differ depending on whether the dataset is captured in a 2D or 3D environment. The tools are easiest accessed through their accelerators (shortcut keyboard buttons) for a more effective workflow. Figure 7.3 displays the tools used during annotation in a 3D dataset, each of which has been numbered and their function explained below:

1. *2D & 3D* - Toggle between 2D view, images captured by cameras, and 3D view, point cloud captured by lidar.
2. *Main Mode & Grid Mode* - Toggle between main mode, showing the point cloud environment, and grid mode, showing the camera, back, top, and right view of the selected object.
3. *Center Mode* - Helps draw a cuboid by placing two points across the center of the object and then a third point, either on the left or right side of the object, to set the width.
4. *Tape Measure* - Measures distances in the point cloud.
5. *Auto-Zoom* - Automatically zoom to selected objects.
6. *Bird's Eye View* - Places the camera straight above the point cloud.
7. *Driver's View* - Shows the selected object from the driver's view of the vehicle that captured the data.
8. *Translate Mode* - Allows the user to move the cuboid in the point cloud or grid mode.
9. *Rotate Mode* - Allows the user to rotate the cuboid in the point cloud or grid mode.

10. *Scale Mode* - Allows the user to change the size of the cuboid in the point cloud or grid mode.
11. *Object Rotation* - Rotates the direction of travel of the cuboid by 90 degrees.
12. *Unify Size* - Applies the same cuboid size of the selected frame to all other frames of the object.

Annotation Process The process of annotating objects in the Kognic Platform consists of multiple steps to achieve an accurate and desired result. The 2D and 3D views are captured by cameras and lidar sensors from a vehicle, referred to as the *ego vehicle*. For each project, there are specific task guidelines that the annotators have to follow, for instance, how accurate the annotations need to be and which objects should be annotated during that task. After entering the task, the annotator begins in the 2D view, see Figure 7.4. This process will not be described in much detail since it is outside the scope of this project. However, important aspects are that the annotators mark out all objects using rectangles or polygons, either in a single-frame or the entire sequence, and determine the class and properties of the object. In the next step, these need to be linked, making each 2D annotation connected to its 3D correspondent. Additionally, some tasks are solely 3D annotating, however, the 2D images are still available for cross-referencing and comparing.

Figure 7.4: 2D view mode while annotating in the Kognic Platform. Vehicles have been marked out using boxes. Screenshot provided with consent from Kognic.

After the 2D annotations are completed, the annotator switches to the 3D view of the point cloud and begins with one of the objects, see Figure 7.5. For single-frame annotations, they look for the object in the 3D point cloud corresponding to the 2D annotation, while for sequences, they also pick the frame where the object is most visible. Finding the correct object can sometimes prove tricky as it is not always clear which object is which. Thus, the annotator utilizes the 2D picture window, which shows a green marker corresponding to the cursor's location in the 3D point cloud. This allows them to see where they are hovering with the mouse. They also utilize cone projection, which is a purple beam in the point cloud obtruding from the ego vehicle. The direction of the beam corresponds to the selected area of the 2D object, allowing the users to find the object in the 3D view. This, along with reference points from the image, such as buildings, trees, and intersections, lets them know when they have found the correct corresponding object.

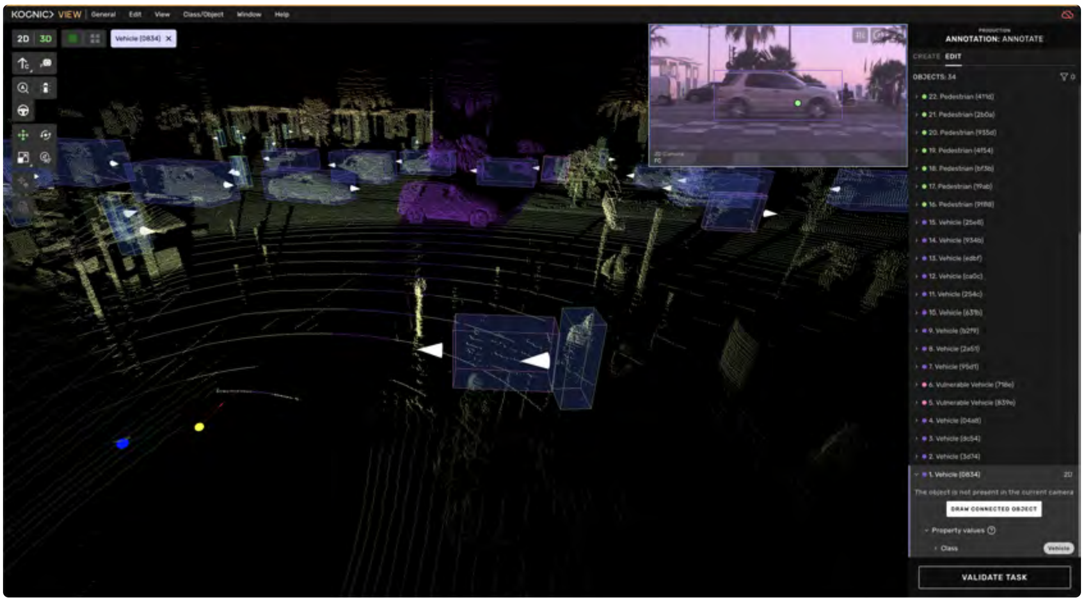


Figure 7.5: 3D view mode while annotating in the Kognic Platform. The vehicle has been located and is to be annotated. Screenshot provided with consent from Kognic.

Next, the annotator selects the Bird's Eye View tool, moving the camera straight above the object in question. Now, the Center Mode tool is selected by pressing the Draw Connected Object button to draw a cuboid by placing two points across the center of the object and then a third point, either on the left or right side of the object, to set the width, see Figure 7.6. If the object is easily distinguishable, the accuracy is high and the direction of the object is correct, if not, an approximation of the cuboid is annotated to be fixed in the following stages.

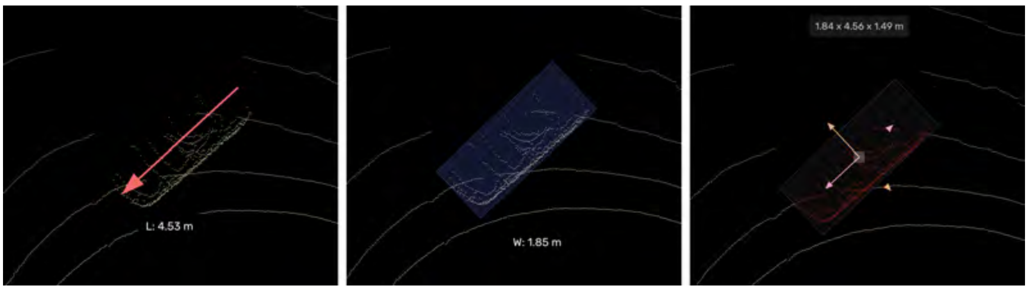


Figure 7.6: The process of drawing the cuboid using the Center Mode tool when annotating a vehicle. Screenshot provided with consent from Kognic.

There are other ways to annotate an object as well. If a cuboid has already been drawn previously, the annotator can copy and paste it to use for a different object. This cuboid has its properties and size already set, making it especially useful for objects of a similar nature. Additionally, for some object classes, the Center Mode tool is assisted with some automation, where the user only has to draw the length of the object and the width is set automatically. Once the cuboid is placed, the annotator enters the Grid View to make adjustments, see Figure 7.7.

Figure 7.7: Grid View mode while annotating in the Kognic Platform. Used while fine-tuning the cuboid. Screenshot provided with consent from Kognic.

The following steps do not occur in any particular order and most tools are utilized multiple times to finetune the cuboid, see Figure 7.8. The annotator decides if the cuboid is in the approximate right location, if not, they either move it using the Translate Mode tool or transform it into position using the Scale Mode tool. To make the rotation of the cuboid correct the annotator utilizes the Rotate Mode tool to make sure it is aligned to the ground and the vehicle itself. To adjust the size of the cuboid, the Scale Mode tool is used allowing the annotator to drag the sides of the cuboid to the edges of the vehicle. It is during this process that micro-adjustments mainly occur. The micro-adjustments in focus are translate, scale, and rotation micro-adjustments, defined as when annotators are making unnecessary adjustments to cuboids of already sufficient quality.

If the direction of the cuboid is wrong, the Object Direction tool is used to rotate the cuboid 90 degrees and is repeated until it is facing the correct direction. They also set the properties of the class, for example determining the visibility of the object and whether it is stationary or moving. If the annotation task is single-frame, the annotator repeats the process for all objects, while if the task is for sequence, they continue using the procedure below.

To complete the annotation of the vehicle, the annotator switches back to the Main

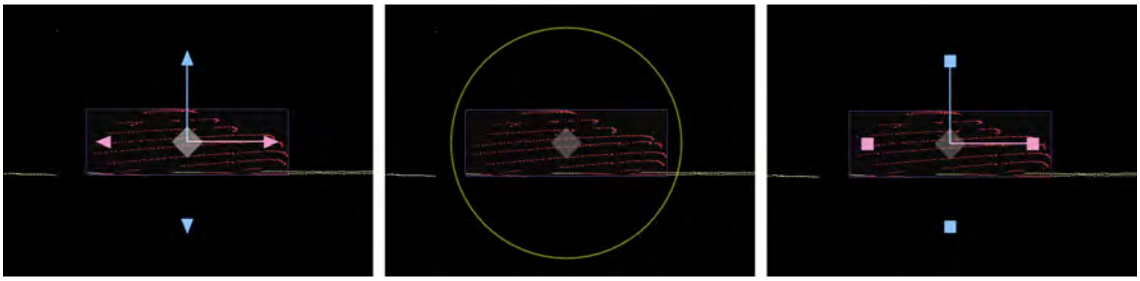


Figure 7.8: Transformation of the cuboid using the Translate Mode, Rotation Mode, and Scale Mode (from left to right). Screenshot provided with consent from Kognic.

View of the point cloud and moves one frame forward in the sequence. Next, they move the object into the new position using the Translate Mode and the Rotate Mode tools, see Figure 7.9. This is repeated for the entire sequence, each time omitting a couple of frames in between. If during any of these frames, the annotators realize that a more accurate annotation can be created, they adjust the cuboid and use the Unify Size tool. This sets the new dimensions of the cuboid for the entire sequence.

Figure 7.9: The process of annotating a sequence. Screenshot provided with consent from Kognic.

The final step in the process is to check if the sequence is complete. The frames the annotator skipped are automatically annotated as the cuboid moves between the annotated frames by itself. They check the sequence to see if the cuboid moves correctly and correlates to the movements of the vehicle. If any small adjustments need to be made for any of the frames, the annotator makes them.

This process is repeated for all objects that should be annotated in the 3D point cloud. Once it is all completed and the annotator is satisfied, the task is submitted.

Review of Annotations

When an annotation is complete and the task is submitted, it gets sent for review. Another more senior annotator or quality manager reviews the work, adding comments and making suggestions for necessary changes. Each suggestion for improvement is a separate feedback item, which also has an error type to more easily identify the problem, for example, a missing object, incorrect dimensions, or ”other”. The reviewer then decides whether the task can approved and submitted for delivery or if it should be rejected and returned to the annotator as a correction task.

If the task is sent for correction, the feedback items containing the comments, suggestions, and error types can be seen by the annotator to make the required changes. The annotator can either resolve the separate feedback items by fixing the problems or reject them, adding comments as to why the annotation is already correct. When this is finished, the correction task gets sent for another review, where the reviewer can once again determine whether it can be approved for delivery or if it needs more corrections. This loop continues until the annotation is of sufficient quality. During this stage, numerous micro-adjustments occur. If the annotator receives an object for correction, they invest additional time to ensure its perfection and prevent it from returning as another correction.

Once the task is submitted for delivery all feedback tasks and respective comments are removed and the annotation gets sent to the customer. The annotations can still be adjusted for a limited period, but someone has to actively go in and review the task as it is otherwise the final step in the review process.

Annotation Statistics

Statistics, being a non-biased source of information, can emphasize the real usage of the Kognic Platform, giving a broader understanding of it. One dataset, including approximately 280,000 cuboids, was provided by Kognic to understand how much time and effort is spent on annotating in their platform. It consisted of information per user regarding their completed tasks and the number of shapes, 2D boxes, polygons, and 3D cuboids drawn, in addition to the amount of time spent in the 2D and respectively 3D views. From this, calculations show that the average time spent in the 3D point cloud per cuboid was roughly 55.43 seconds.

With another dataset provided by Kognic, the average number of adjustments of a cuboid was calculated to be approximately 17.16 times. This number of adjustments came from a random sample of around 500,000 cuboids. The measurement started after the annotator had drawn the initial cuboid using the Center Mode tool until they began with the next object.

When exploring the dataset, annotations with only one or two adjustments were deemed inaccurate and therefore removed from the dataset. This was decided since two or fewer adjustments are insufficient to create a correct cuboid. These annotations were probably mistakes, therefore after the creation of the cuboid, they were simply deleted. This could have been because the cuboid was drawn in the wrong position or that the object should not have been annotated in the first place.

Another interesting dataset provided by Kognic was the number of adjustments on a cuboid after being duplicated instead of being created from scratch. The average number of adjustments turns out to be approximately 11.08 times. This quantitative measurement gives another perspective on creating cuboids and different annotation processes. The data gathered in this section will later be analyzed in Section 7.1.6.

7.1.4 User Research

The user research collected during this Master's thesis was from three different methods, observations, semi-structured interviews, and a workshop see Section 4.3, 4.6, and 4.7 respectively. Each was used to get valuable information from different stakeholders, where the observations, semi-structured, and workshop concerned annotators, expert users, and Kognic respectively. This leads to a broad understanding of the problem area from different perspectives.

Observations

Observation videos were used to gain an understanding of the naturalistic behavior of annotators using the Kognic Platform. The observation videos showed single-frame point cloud annotations, following some instructions specifying what content to include in the recordings. The guidelines asked for six video recordings, each lasting at least 10 minutes. Additionally, the annotators were instructed to start the recording midway through their task, to get a more naturalistic behavior of their workflow. In total, nine observations from three different annotators were received, each approximately 10 minutes long.

The observations were analyzed and the workflow of annotating was deconstructed and written down. Interesting behavior regarding micro-adjustments was also examined, the result from the observation is presented in Section 7.1.5.

Semi-structured Interviews

Two semi-structured interviews were conducted with expert users, each with multiple years of experience. These users now have the position of quality managers, where they review and make corrections to finished annotations and help with ways of working in the Kognic Platform, creating videos, instructions, and supporting annotators. The interviews were each one hour long and the audio was recorded to later be transcribed and analyzed. Additionally, during the interview, the users were sharing their screens showing how they annotate using the Kognic Platform. This helps them to answer the questions since it acts as a medium to talk about.

The interview consisted of general questions regarding working with annotations, combined with narrow questions aimed at investigating micro-adjustments. Since the research goal of this project is of a wicked nature, the final design solution is yet to be determined. Therefore, the interviews were not solely focused on micro-adjustments. Instead, the purpose of the interviews was to generate a deeper understanding of the annotator's work, exploring the essence of the occupation, the process of annotating, and gathering insights about the cause of micro-adjustments,

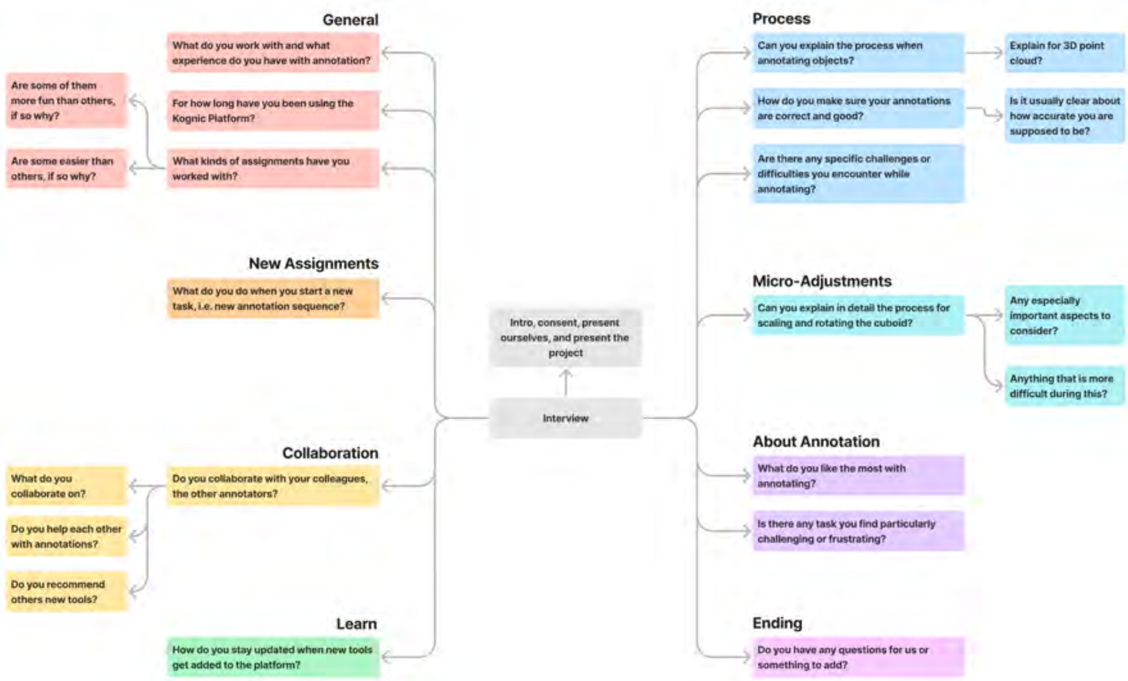


Figure 7.10: An overview of the questions asked during the semi-structured interviews.

see Figure 7.10 . Finally, the interviews were transcribed, where interesting and relevant insights were cited from the transcriptions. The results of the interviews are presented in Section 7.1.5.

Workshop

The workshop was conducted to gain insights about micro-adjustments from six members of various teams at Kognic. It was one hour long and held online using a *Miro* [95] board as it was deemed the easiest way to gather everyone and for more effective collaboration. The workshop was held in English and divided into three segments, first a quick icebreaker, next an introduction of the purpose of the workshop, and finally, information gathering.

During the icebreaker, everyone briefly introduced themselves and what they worked with at Kognic. The next part consisted of finding specific icons, revealed one at a time, in a cluster of different emojis. This was used to get the participants into an exploratory mindset and ease them into the workshop. The second phase was the introduction of its purpose. The aim being an exploratory workshop to gather information about the underlying problem for micro-adjustments from employees at Kognic, gaining new insights into the problem area.

The information gathering phase, the major chunk of the workshop, began with playing a short one-minute video of an annotator making several micro-adjustments to a cuboid. This was shown to remind the participants of the problem area and get their ideas flowing. Next, a ten-minute individual braindump session was introduced, where the participants were to write down anything they could think of on Post-it notes. If they got temporarily stuck, there were prompts available to help them along the way, which were ”Why it occurs?”, ”How it occurs?”, and ”Where have you seen it?”. After this, the participants were separated into two breakout rooms working in groups of three with one Master’s thesis student in each room to assist and keep everyone on track. If a Post-it did not have a clear underlying problem written down, it was reformulated to include one. The participants categorized their ideas into groups of similar kinds and put them into one of four areas: problems I have observed, problems based on feedback, individual thoughts, and others. If they came up with any new ideas during this phase, they were encouraged to also write those down.

The workshop concluded by gathering both groups again and explaining their most important findings to the others. The analysis and results of the workshop can be found in Section 7.1.5.

7.1.5 Affinity Diagramming

This section will describe the analysis and results of the observations, semi-structured interviews, and workshop. To synthesize the gathered material, three affinity diagrams, see Section 4.8 were created, one for each user research method. The affinity diagrams were created in Figma.

Observations

While watching the videos, interesting observations of what annotators did while annotating were written down. These were then transferred onto Post-it notes in Figma, to be categorized and grouped using affinity diagramming. The four main areas that were deemed important were *ways of working*, *rotation*, *perfectionism*, and *micro-adjustments*. These categories are explained below and the affinity diagram can be seen in Appendix A.1.

Ways of Working The ways of working theme included observations showing that rotating the cuboid before scaling it was more efficient than the other way around. The reasoning for this was that if the annotator started with scaling, they would need to adjust the size again after completing the rotation. The observations showed that some annotators started with scaling, despite being the less efficient approach. Important to note, that if the cuboid initially was inaccurate, some transformation would be required first for it to be possible to align it to the ground.

Another insight concluded from the observations was that the Translate Mode tool was rarely used, instead, the annotators changed the position through the Scale Mode tool, scaling the box into position. The only use case of the Translation Mode tool was if the cuboid was oddly far away from the object or when moving the cuboid between frames in an annotation sequence.

Rotation The rotation category included observations of users having difficulty adjusting the rotation of a cuboid. This included some annotators rotating the cuboid just slightly, very fast, multiple times to align it. Another aspect observed was different users' solutions for aligning the cuboid horizontally to ground level, some utilized the edge of the window as a guideline, others increased the width of the cuboid to get a larger base to use as a reference.

Perfectionism Perfectionism was another aspect which was observed during the observations. For example, even though a cuboid was already of sufficient quality, some annotators made additional adjustments to double-check their work. This was done by changing the size of a perfectly annotated cuboid just to see if it was sized correctly and then putting it back in the same position.

Micro-adjustments The observations also generated an understanding of where in the annotation process micro-adjustments mainly occurred. This was after the user entered the grid mode while transforming the cuboid and especially showed that adjusting the size and rotation of the cuboids was an issue. Additionally, observations showed that objects with a large number of points were often more prone to micro-adjustments than objects with only a few.

Semi-structured Interview

The affinity diagram for the interviews began with writing the quotes extracted from the transcriptions of the interviews on Post-it notes. These were iteratively categorized and clustered, creating a structure to make a large amount of information comprehendible. The most important categories, *ways of working*, *perfectionism*, *micro-adjustment*, *expert users*, and *task guidelines* are described below and the entire affinity diagram can be found in Appendix A.2.

Ways of Working The ways of working category includes how expert users work towards finding the most efficient work process:

> *"I try different ways of annotation, and I check which is faster, or I reach out to colleagues that I think have more experience than me in this area, and ask them for help."*

The quote implies that expert users are prone to find the most efficient process, taking help from others to do so.

Perfectionism Perfectionism as a concept was never explicitly brought up during the interviews, however, its presence could be inferred between the lines. For example, when talking about the quality of cuboid one interviewee said:

> *"If my goal is to create the perfect annotation, I can do even some really small adjustments, like move it for a few millimetres."*

This statement, as well as the correlating footage, indicates that some adjustments are solely made to make the cuboid perfect, not because it is needed.

Micro-adjustments Notes regarding rotation explain how the annotators are aligning the cuboid to the ground. When one of the interviewees demonstrated how rotations were done correctly, they said:

"Maybe I repeated it too much."

The quote is interesting since the goal was to show how a cuboid is annotated correctly, but when doing so the interviewee performed micro-adjustments.

Expert Users The interviewees showed that the work of an expert user is different from that of an annotator. Their work is focused on providing instruction for annotation, reviewing already annotated tasks, and making changes to existing annotations, among others.

Knowing there is a distinction between them is important to acknowledge during this project since they might have different perspectives and work processes. Thus, both stakeholders are using the platform, and no side should be neglected in the redesign.

Task Guidelines One aspect that was highlighted in the affinity diagram was that it is easier to follow the task guidelines for 2D than 3D annotations. This is because 2D images are pixel-based, allowing for a precise boundary of what is sufficient or not, whereas the 3D point cloud is vector-based, which makes it impossible to have such guidelines. The quote below outlines this problem:

"In 3D, I think we have some similar [how many pixels are allowed between the line and the object], but there are no clear requirements because there are no pixels or something like that."

Another aspect that was brought up from the affinity diagram was that objects further away are less important than objects close to the ego vehicle. The objects close to the ego vehicle will include more points, allowing for a more precise encapsulation. This insight is further reinforced by this quote:

"The further away the object is, the bigger mistake you can make. It will be acceptable because, from this distance, it's impossible to say if it's actually a correct level."

Workshop

The resulting Post-it notes from the workshop were exported from Miro into Figma to further analyze the information using affinity diagramming. The Post-its were sorted and put into different categories of various kinds. The results show a lot of information about micro-adjustments, from where they occur to where they could originate from. The most relevant and important categories are explained below, all of which to consider for the remainder of the design process and the creation of the requirements specification. The entire diagram with all the categories can be found in Appendix 7.1.5.

Culture The annotators' culture is naturally embedded into their work. Their work ethics are high and they want to create perfect annotations of utmost quality. This topic of culture was one of the categories discovered during the affinity diagramming. The following two Post-it quotes from the workshop outline this problem:

"There is a cultural pride observed in delivering good quality."

"They have a culture that the boxes should be of the highest possible quality."

Ways of Working Another theme that emerged from the workshop was the ways annotators work. There is no right or wrong way to annotate and many users approach the task differently, even between objects. Below, some quotes from the Post-it notes regarding this topic are cited:

"Observation: The fastest are often also high-quality Theory: Maybe they are fast precisely because they know what is sucient."

"Wrong workflow leads to unnecessary work."

"We know that the final bit [of annotation] is fairly ambiguous. If we ask many annotators to draw the same box they reach quite different results."

The workshop helped to further understand that there are more efficient ways to annotate than others. Different workflows lead to different results and some lead to unnecessary work.

Uncertainty and Feedback Uncertainty and feedback were another two categories that surfaced during the affinity diagramming of the workshop. Though they are separate categories, they are directly related to each other. Through insufficient feedback, annotators have a difficult time knowing when an annotation of an object is good enough. Two quotes from the workshop about this situation are the following:

"There is no indication of when a cuboid is good enough."

"[There is] no clear "finished" indicator, [the annotators are] trying to make it perfect."

Diminishing Return The diminishing return of annotation was also a category that came from the workshop. For annotations, the diminishing return is a point on an inverse exponential function graph where the number of adjustments reaches the required quality of the cuboid, see Figure 7.11. Any additional adjustments lead to unnecessary work while fewer resulted in insufficient quality. This is a difficult point to determine and can change from task to task. However, the problem stays the same, the annotators do not know when they reach a sufficient quality of the cuboid. The following quotes from the workshop outline this problem:

"If they knew perfectly well what the required quality was they would not overproduce."

"The first bit of adjustments brings the most value, the longer it goes the smaller the adjustments become and the less valuable they are."

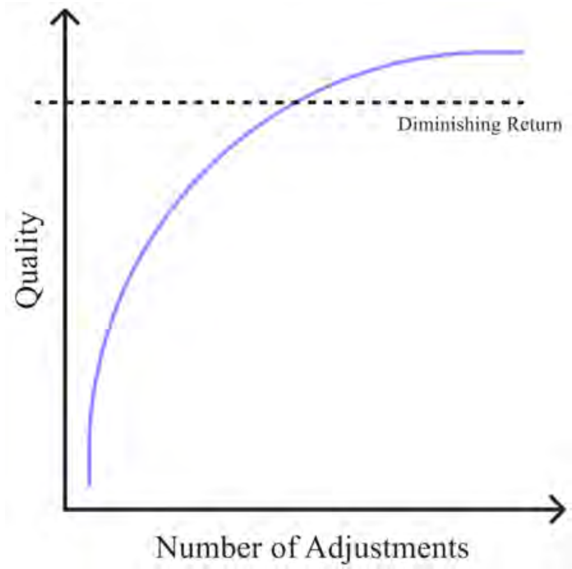


Figure 7.11: An approximation of the diminishing return graph, when the number of adjustments reaches the required quality of the cuboid.

Perfectionism The theme of perfectionism was another topic of importance in the workshop. This is also a category that is closely related to the others mentioned above: culture, uncertainty, feedback, and diminishing return. Annotator's culture and pride make them want to create perfect annotations. The uncertainty and lack of feedback also result in perfectionism, as they do not know when a cuboid is good enough and therefore make it perfect. Two participants during the study described perfectionism within annotation very well, with the following citation from their Post-its:

"This was a very visible object - with a lot of effort from perception to result. Like eating with chopsticks."

"The users think they're contributing with useful actions by making it absolutely 'pixel perfect' meanwhile it doesn't matter to the annotation by spending time on all the minute details."

Tools The tools category also emerged from the workshop. It was discussed that some tools could be improved to help the annotation process, for example, the drawing tool could not produce an accurate enough initial result or that it does not give the best possible view at all times. However, another insight from the participants was that improving the tools is not always the right solution, sometimes there could be other underlying issues at hand.

Number of Adjustments The number of adjustments was another subject discussed during the workshop. The annotators are given a target time for each annotation but not for the number of adjustments. The following citation highlights this problem area:

> *"As an annotator, I am given the option to do any number of edits without repercussion."*

Rotation Of all types of micro-adjustments, the Rotation Mode tool was by far discussed the most. There were many problems analyzed with the rotation of cuboids. Making them level with the ground and aligning with the sides of the object was difficult due to the ground as a reference point being a challenge and vehicles having curves. Two quotes from the Post-its highlight this especially well:

> *"Feedback from annotators is that making things level with ground is difficult."*

> *"Hard to align with a side of an object due to no straight sides of an object."*

Automation The theme of automation was also brought up as having a relationship with micro-adjustments. Usually, automation tools, such as the snapping tool are not used. The snapping tools allow the cuboid to snap to the outermost data point of the object which is useful when it can be accurately used. Additionally, the annotators are not always aware of their existence. With the use of automation, micro-adjustments could be reduced. However, as mentioned earlier, annotators feel pride in creating high-quality annotations. This could lead to still making micro-adjustments after automation is used because they want to put their own effort into it.

7.1.6 Task Analysis

Based on the literature review about task analysis for HCI, see Section 3.4, it assists in fully understanding the process of annotation, also enabling the comparison of data. During this Master's thesis, three different methods were utilized, flowcharts to get an understanding of the task, KLM-GOMS to get quantitative data measured in time, and an optimal versus actual comparison of the amount of adjustments to get quantitative data measured in clicks. This section presents how the methods were utilized and their respective results, each giving important insights into this Master's thesis.

Flowchart

A flowchart was created to generate an understanding of all aspects of annotation during a task, see Section 4.4. This is useful during the ideation phase to stay in the scope of the project while still considering what needs to be included in the process. This is done by breaking down the task into subcategories and showing where in the process the annotator makes decisions. Since the observations showed that micro-adjustments mainly occur after entering the grid view when the cuboid is rotated and scaled, this part of the process is of most interest. Figure 7.12 illustrates that part of the annotation process and the entire flowchart can be found in Appendix A.4.

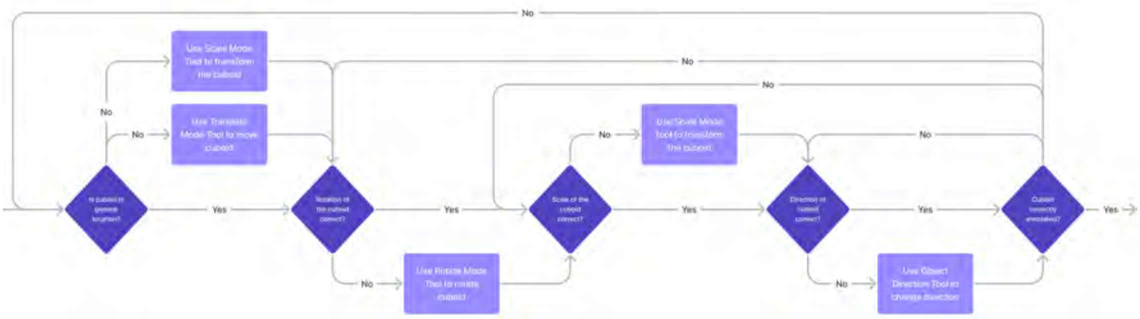


Figure 7.12: Flowchart of the part of the annotation process where the majority of micro-adjustments occur.

Adjusting the cuboid is an iterative process where the annotator constantly goes back and forth utilizing the various tools several times. To make the flowchart easier to understand and interpret, it was simplified to a more linear process. However, in reality, all decision points in the flowchart could happen in different orders from annotation to annotation.

KLM-GOMS

The KLM-GOMS method further helps show the decomposition of the general tasks of annotating, showing each step of the process in extreme detail, see Section 4.5. The method will also allow for a comparison of the optimal time and the actual time for annotation, resulting in quantitative data on efficiency. Not all aspects of the annotation process are interesting to analyze, only the part where the micro-adjustments occur, shown in the segmented flowchart, see Section 7.1.6. It is this sequence of the annotation task that was analyzed and the execution of the method can be seen in Table 7.1.

With the decomposition of general tasks for the sequence where most micro-adjustments have been observed, one can calculate the optimal time it takes for an annotator to complete the task using the equation below, where n is the number of times the unit task has been executed.

$$T = (K * n) + (P * n) + (B * n) + (M * n)$$

$$T = (0.28 * 3) + (1.1 * 22) + (0.1 * 26) + (1.2 * 2)$$

$$T = 30.04s$$

Table 7.1: The decomposition of general tasks for annotation in the Kognic Platform, with a focus on the sequence where most micro-adjustments have been observed.

Type	Time (Seconds)	Description
P	1.1	Point to Draw Connection button.
BB	0.2	Click mouse button on the Draw Connection button.
P	1.1	Point to the middle of the back of the object.
BB	0.2	Click mouse button.
P	1.1	Point to the middle of the front of the object.
BB	0.2	Click mouse button.
P	1.1	Point to either the left or the right side of the object.
BB	0.2	Click mouse button.
K	0.28	Press 'G' keyboard button to enter Grid View.
K	0.28	Press 'X' keyboard button to enter Rotation Mode.
M	1.2	Mental act of routine thinking.
Repeat the next four tasks three times (back, top, and right object view).		
P	1.1	Point to rotation grab point.
B	0.1	Press the mouse button.
P	1.1	Drag mouse to rotate object.
B	0.1	Release the mouse button.
K	0.28	Press 'C' keyboard button to enter Scale Mode.
M	1.2	Mental act of routine thinking.
Repeat the next four tasks six times (top, bottom, left, right, front, and back sides).		
P	1.1	Point to scale grab point.
B	0.1	Press the mouse button.
P	1.1	Drag mouse to scale object.
B	0.1	Release the mouse button.

This optimal time can then be compared to the actual time it takes for annotators to complete one annotation. However, occasionally, the actual time can subseed the optimal time. It is difficult to decipher the exact reason behind this. It could be because some angles and cuboid sizes are perfect from the start and therefore adjustments to the annotation are not needed or the annotations could be created fast but be of insufficient quality. Thus, the optimal time calculated is still a good measurement to use as a basis for comparison because it will be both an accurate and fast annotation time, considering all potential adjustments needed. To calculate the actual time, thirty randomly selected annotation times are recorded from the observation videos of three different users, see Table 7.2.

Table 7.2: The actual time it takes for annotators to complete one cuboid annotation.

	Time per Annotation of Cuboid (Seconds)									
User 1	59	83	48	66	71	75	88	65	56	80
User 2	32	26	22	43	32	42	25	20	56	50
User 3	38	51	32	49	26	50	63	51	57	24

From some basic calculations, the average time for the annotators to complete the task is 49.33 seconds. This, in theory, shows that it is possible to save up to 18.93 seconds per annotation by reducing adjustments, a reduction of 39%. When considering the large amount of cuboids created per day, it would save a lot of labor. These statistics are relevant to this Master's thesis since a decrease in the number of micro-adjustments, even by a small amount, could reduce the time spent on annotations significantly. The potential of lowering the number of adjustments will be analyzed in the next section.

It can be assumed that the average time for annotators to complete the task based on the observations is quite accurate. As mentioned in Section 7.1.3, the total time annotators spend in the 3D point cloud per created cuboid was calculated to be approximately 55.43 seconds. Thus, annotators spend around 6.10 seconds looking for each object before they start drawing the cuboid. This is a reasonable number as sometimes they spend a longer amount of time, for example, if it is the first object in the task to be annotated, and sometimes they spend a shorter amount of time, for example, if the objects to be annotated are located close to each other.

Optimal vs. Actual Number of Adjustments

With the KLM-GOMS method giving results in the metrics of time, a comparison of optimal versus the actual number of adjustments can give a result in the metrics of quantity. This would further support the purpose of this Master's thesis and give a quantitative estimate of the number of adjustments annotators make.

The optimal number of adjustments is nine, rotating three times and scaling six times, which is a good measurement to use as a basis for comparison because it considers all potential adjustments needed. When comparing this to the actual number of adjustments, see Section 7.1.3, annotators make on average 17.16 adjustments to a cuboid, which is 8.16 more adjustments than what could be possible to achieve the task. Some of these adjustments would be necessary to make an accurate annotation while others would be considered micro-adjustments. This quantitative data further supports the result of the KLM-GOMS method by showing that more adjustments are made than the optimal amount. If these could be reduced, time would be saved during annotation.

If a cuboid is duplicated instead of being created from scratch, the average number of adjustments is approximately 11.08 times, see Section 7.1.3. This method of annotation is more efficient, with a reduction of 6.08 adjustments per cuboid than if it was created by using the Center Mode tool. This data shows that duplicating cuboids is better in lowering the number of adjustments. This process is limited to objects that have a similar nature, such as size and properties. If that is not the case, copying objects could potentially lead to more adjustments, since many changes would be required.

7.1.7 Requirements Specification

The process of creating the requirements and guidelines specification started by eliciting insights from the user research deemed as important through affinity diagramming. To prioritize and organize these insights, the MoSCoW method, see Section 4.9, was utilized. The important categories gathered from the affinity diagramming were rewritten into requirements or guidelines statements supporting the final design solution. They have then been categorized and prioritized into *must have, should have, could have,* and *won't have*, see Table 7.3.

7.1.8 Guidelines Specification

When creating the guidelines for how to reduce micro-adjustments during annotation in 3D platforms, all the categories from the requirements specification must be considered. Some categories which resulted from the affinity diagramming do not apply to the final design, therefore, they were not added to the requirements specification. However, these apply to the guidelines and are important to consider when creating them. This section presents a list of the remaining specifications not included previously:

- The guidelines should consider the feedback structure throughout the annotation process and align all stakeholders.
- The guidelines should consider the annotators' culture.
- The guidelines should consider the process of reviewing annotations.

Table 7.3: The requirements specification developed using the MoSCoW method.

Must Have	The final design must reduce the number of micro-adjustments.
	The final design must allow the annotator to create both perfect and good enough annotations when necessary.
	The final design must not increase the number of annotations with insufficient quality.
	The final design must not interfere with the carrying out an annotation process.
	Annotators must feel like they are contributing with their work.
Should Have	The final design should reduce the number of rotation micro-adjustments.
	The final design should reduce the number of scale micro-adjustments.
	The final design should not interfere with the feeling of being in control.
	Feedback should be provided to show when a cuboid is good enough and further annotation is unnecessary.
	The final design should help annotators to adapt the quality depending on the project's guidelines.
	Annotators' pride in wanting to create perfect annotations should be considered in the final design.
	The annotation tools should be improved to minimize micro-adjustments.
	The final design should utilize automation when possible.
	The final design should minimize perfectionism.
Could Have	The final design could provide knowledge to annotators of when the diminishing return of a cuboid occurs.
	The final design could provide feedback on the annotation process.
	The final design could provide help to align the box to the sides of the object.
	The final design could provide help to align the box to the ground.
	The final design could include clear 3D accuracy requirements.
	The final design could provide a structured process for annotating an object.
Won't Have	Provide an easier training and onboarding process.
	Reduce the pressure of needing to perform perfect annotations.
	Annotators should be provided with a culture that minimizes perfectionism.

7.2 Phase 2: Sketch and Decide

This section explores the sketch and decide phases of design thinking, see Figure 7.13. It will present the results of the ideation process and concept evaluation, where a limited number of developed ideas will be used as a basis for the third phase and final design.

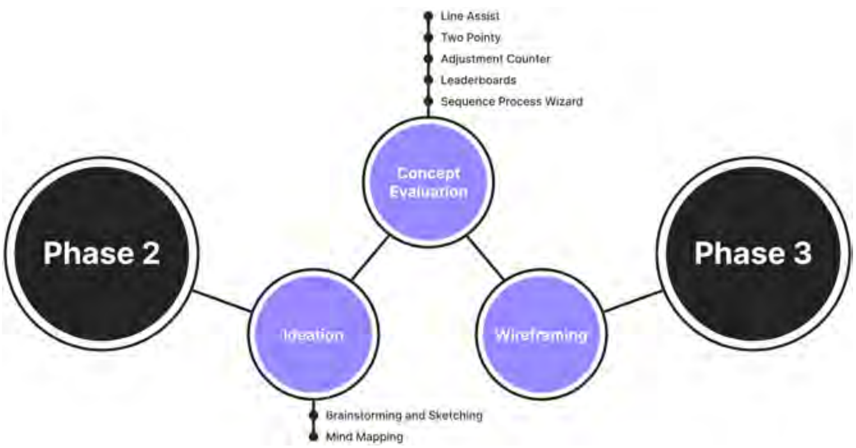


Figure 7.13: Phase 2: Sketch and Define.

7.2.1 Ideation

The ideation process, see Section 4.10, utilized several different interaction design methods, outputting a large number of ideas to fully explore the solution space. Using these various methods allowed the ideation process to cover all topics and areas of interest, which resulted in a broad range of solutions being generated. Supporting the ideation is the requirements specification, where each requirement or guideline was a topic for inspiration and in need of a design solution. The ideation methods used during this Master's thesis were brainstorming, sketching, and mind mapping, each process and their respective results are explained below.

Brainstorming and Sketching

The brainstorming session revolved around finding solutions and coming up with ideas for the various topics in the requirements specification. The session was open and unrestricted, allowing for the exploration of all ideas that come to mind. Post-it notes were used to either write down or sketch the ideas, see Figure 7.14, enabling a common understanding of the concepts and eliminating potential misunderstandings. Keeping it visual also allowed for easier inspiration gathering and dialogue during the session.

Two 45-minute brainstorming sessions were held where the Master's thesis student ideated on the topics, followed by brief discussions. After the final discussion, a more extensive review was conducted, further developing and exploring the ideas. The result of the brainstorming sessions was a comprehensive list of numerous concepts to later be evaluated.

Figure 7.14: Image of the brainstorming process, where ideas were written down or sketched onto Post-it notes.

Mind Mapping

After the brainstorming session, most of the resulting concepts were focused on single-frame annotation. To further explore sequence annotations, discussions between the Master's thesis students were held regarding which areas should be considered. The chosen areas were written down and further broken down into smaller elements, including design ideas and other aspects of importance. This resulted in a mind map, see Figure 7.15, displaying a comprehensive image of how micro-adjustments in sequence annotations could potentially be reduced.

Discussions on how to further develop the ideas from the mind map into functioning concepts were held, during which sketches and post-its notes were created to establish a common understanding of the ideas. The materials produced during the discussion were later combined into a final list of various concepts to be evaluated.

7.2.2 Concept Evaluation

To evaluate the concepts, a couple of different approaches were used to isolate the most promising ideas to develop further and focus on. One of these methods was meeting with a developer at Kognic to evaluate which ideas were possible to implement. This also resulted in learning that some of the concepts had already been tested previously and proved to be ineffective, in addition to that a couple of the concepts were already under development. Another evaluation was with the Master's thesis advisors at Kognic, being UX designers at the company, who gave input

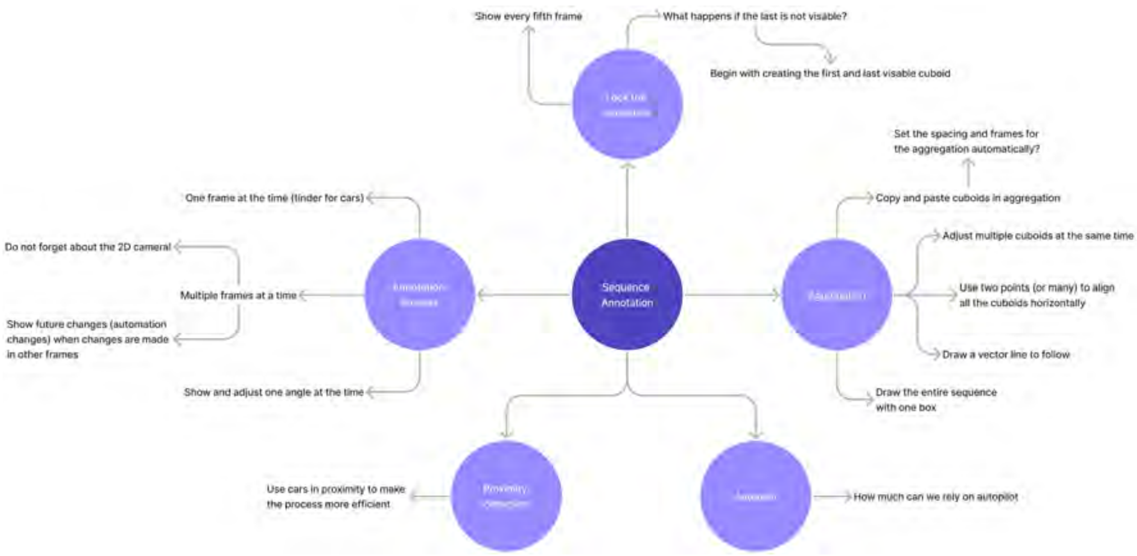


Figure 7.15: Mind map for the ideation process of sequence annotation.

into which ideas they thought were more desirable than others. This resulted in feedback and suggestions for improvement to further explore and contemplate.

The final evaluation was done by the Master's thesis authors, writing pros and cons for each concept and voting on which to continue with for the next phase of the design process. The voting was done with the feedback from the developer and advisors in mind, together with the knowledge gathered throughout the project. The resulting concepts to further explore and develop were *Line Assist*, *Two Pointy*, *Adjustment Counter*, *Adjustment Leaderboard*, and *Sequence Process Wizard*. These concepts are explained in further detail below.

Line Assist

Line Assist is a concept to help reduce the rotation micro-adjustments seen during the observations. Extruding from the cuboid are dashed lines to help the alignment process, both horizontally to the ground and to the object itself.

Two Pointy

Similar to Line Assist, Two Pointy also aids the rotation alignment of the cuboid, reducing the number of micro-adjustments. The concept revolves around the user placing two points on ground level, horizontally aligning the cuboid to the ground.

Adjustment Counter

The Adjustment Counter is a concept that gives feedback to the annotators on the number of adjustments made per annotated object. The counter increases with each adjustment, nudging the user into wanting to complete the task with a lower number of adjustments. Additionally, the concept has the potential to further nudge the annotator into reducing their micro-adjustments by changing the color of the counter from green to yellow to orange to red as the counter goes above the optimal number of adjustments.

Adjustment Leaderboard

This concept aims to reduce the number of micro-adjustments through gamifying the annotation process by including some sort of leaderboard. This could be done through a high score list amongst colleagues, displaying the average number of adjustments per cuboid throughout each workday. Another idea is to have an individual high score list, displaying some sort of message each time the annotator reaches a lower number of adjustments for a cuboid.

Sequence Process Wizard

To help the annotators achieve an efficient workflow, the concept of a Sequence Process Wizard was developed. This concept aims to lower the number of micro-adjustments by indicating to users what tools to use and when. The wizard will also indicate when a tool has already been utilized, which potentially could nudge the users to only use each tool once. This concept guides the users through the process of annotating a sequence, following the optimal workflow.

7.2.3 Wireframing

Wireframing, see Section 4.11.1, was utilized to explore the evaluated concepts further, establishing a fundamental structure to the designs. The concepts were wireframed on paper using a quick and dirty approach, allowing for a rapid and iterative prototyping session, as well as reducing the risk of getting too attached to the ideas. An overview of the outcomes from the wireframing process can be seen in Figure 7.16.

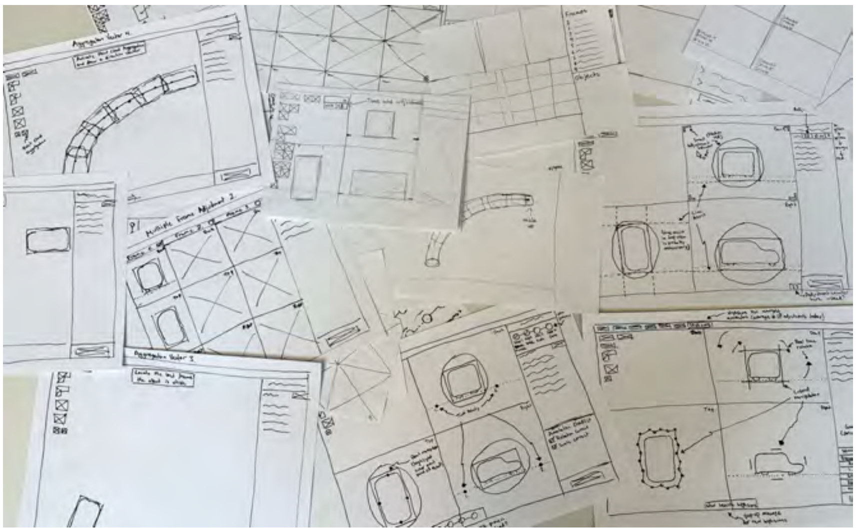


Figure 7.16: An overview of the results from the wireframing process.

The results of the wireframing process gave an understanding of how the various concepts would look and feel, in addition to the overall layout and positions. Together with the description of the ideas, the wireframes were used as a basis for the initial prototyping, which can be read about in Section 7.3.1.

7.3 Phase 3: Prototype and Evaluate

This section explains the process and execution of the prototype and validate phases of design thinking, see Figure 7.17. It was an iterative phase with several redesigns based on feedback and various evaluation methods. The outcome of this phase will be the result and final design for this Master's thesis presented in Section 9.

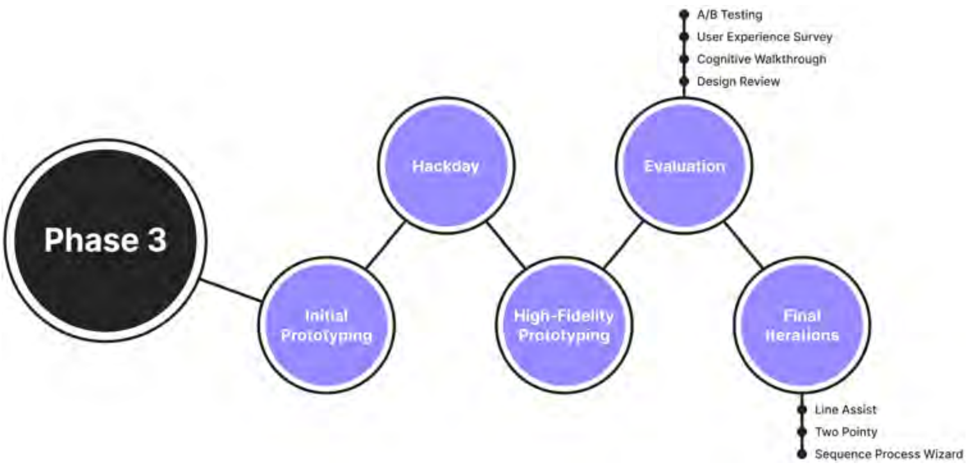


Figure 7.17: Phase 3: Prototype and Validate.

7.3.1 Initial Prototyping

To further develop the wireframes, simple prototypes were created in Figma. These prototypes were built on images from the Kognic Platform to give more context to the concept. Additional information including specific functions and more context behind the ideas was written down on Post-it notes. This was done to get a better understanding of the concepts and to act as a basis to start the initial development, see Section 7.3.2. The initial prototypes for Line Assist, Two Pointy, Adjustment Counter, Adjustment Scoreboard, and Sequence Process Wizard can be seen in Figures 7.18, 7.20, 7.21, 7.19, and 7.22 respectively.

7.3.2 Initial Development

Three members from one of the development teams helped with refining and programming some of the concepts from the initial prototyping. This development occurred over one day and allowed for testing in the Kognic Platform itself, investigating potential aspects for iteration before high-fidelity prototyping and user testing.

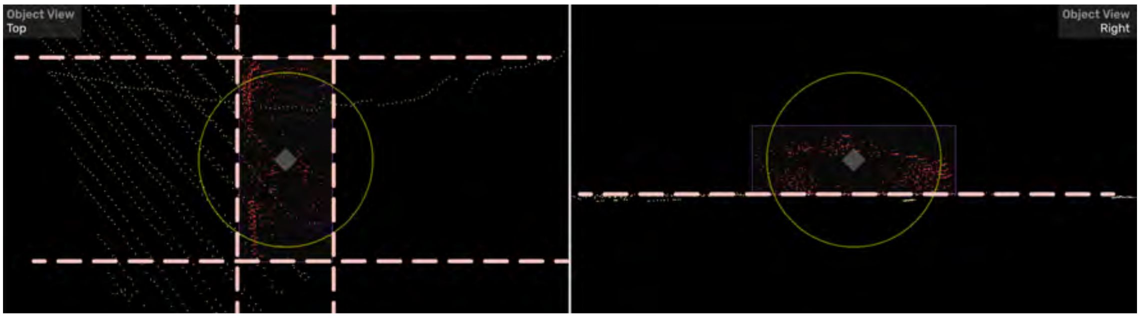


Figure 7.18: Initial Prototype of Line Assist.

Figure 7.19: Initial Prototype of Adjustment Scoreboard.

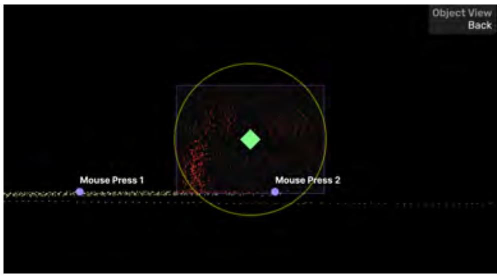


Figure 7.20: Initial Prototype of Two Pointy.

Figure 7.21: Initial Prototype of Adjustment Counter.

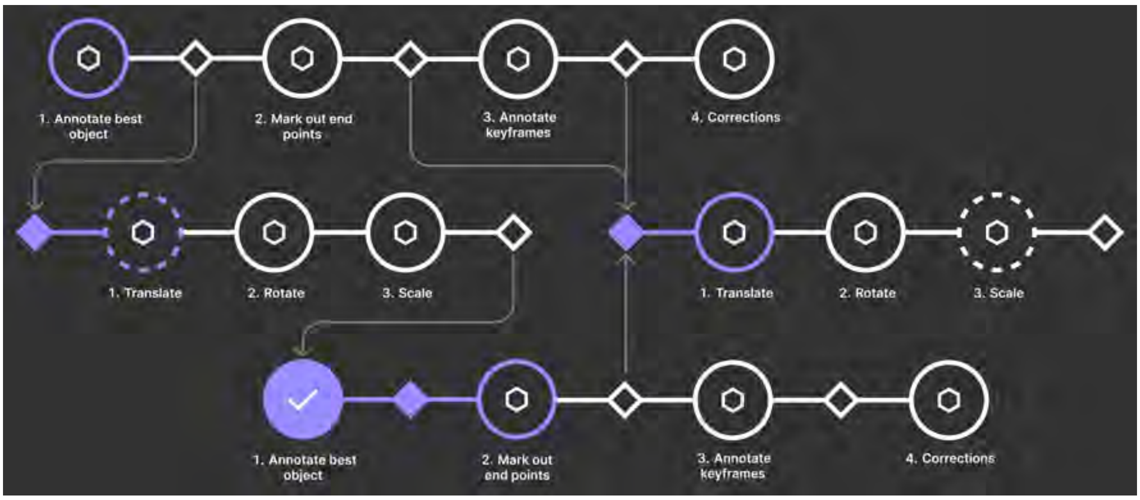


Figure 7.22: Initial Prototype of Sequence Process Wizard.

The two concepts in focus were Line Assist and Two Pointy, both being developed for the testing environment of the platform. These concepts were chosen together with the developers and the thesis authors since they appeared fun and possible to develop within one day. Additional discussions about the Adjustment Scoreboard, Adjustments Counter, and Sequence Process Wizard were also held. The feedback presented on the initial prototype of the Adjustment Scoreboard was that no additional icons or text should be added to the class list. This was because more information in that section would lead to a cognitive overload. Additionally, the Sequence Process Wizard could also be adapted for single-frame annotation, but the order of operations would need to be considered as it would differ from sequence annotation. The final feedback was that the Adjustment Counter had already been developed but not further explored or tested. Additionally, it was brought up that the Adjustment Counter would be a valuable measurement to track performance and that a clear adjustment goal during annotation could potentially increase efficiency.

The Line Assist concept was developed, following its low-fidelity prototype, see Figure 7.23. It featured dashed lines extruding from the base of the cuboid in both the back and side object view windows, as well as dashed lines protruding from all sides in the top object view window. Additional features that were added to this concept were the ability to snap one cuboid to another cuboid's base. This was done to allow for easier horizontal alignment to the road if the two cuboids are in proximity to each other on the same surface. The feature works by holding down the shift keyboard button and translating the cuboid in the z-axis, which aligns it to the nearest cuboid's base.

Figure 7.23: The developed Line Assist concept.

The Two Pointy concept was also developed, having many resemblances with its low-fidelity prototype and only including some minor changes, see Figure 7.24. Instead of marking out two points on the road to rotate the object, the user clicks on one point, holds, and drags out a line, while holding down the shift key. The cuboid's rotation was then adjusted relative to the line, allowing the user to rotate the object in real time by changing the angle of the line, moving it upwards or downwards. The pivot point of the cuboid was the first point that was pressed. The concept permits the user to either use the ground to drag out the line or drag it anywhere in the window, enabling the possibility of using the surrounding environment when aligning the cuboid to the ground.

Figure 7.24: The developed Two Pointy concept.

7.3.3 High-Fidelity Prototyping

After the initial prototypes and development had been made, the Line Assist, Two Pointy, Sequence Process Wizard, Adjustments Counter, and Adjustments Scoreboard were further developed into high-fidelity prototypes. This was done for the prototypes to convey the final look and feel of the design, allowing further exploration of concept details such as the precise colors, position, and animations. The high-fidelity prototyping began with implementing any iterations necessary discovered during the initial prototyping and development. Subsequently, each concept was developed to follow Kognic's design system, using its colors, fonts, and components.

Different versions of the concepts were created, testing small variations of the iterations and comparing them to each other, finding the optimal design. For Line Assist, the main parameters investigated were the color and width of the line, see Figure 7.25. This resulted in the decision made by the Master's thesis students to make the line the same color as the cuboid itself, which is dependent on the class. It was also decided that the width should have the same line width as the cuboid. Another aspect that was iterated during the high-fidelity prototyping was the removal of the assisting dashed lines in the top view window. These were simply unnecessary as they neither helped nor guided the annotator in any way. Finally, the snapping of the cuboid to the nearest cuboid's base was removed from the concept. When this feature was tested by the Master's thesis students, it was discovered that it was not accurate enough since small bumps and level changes in the road led to the cuboid not being aligned correctly. This resulted in additional adjustments needed for many of the cuboids.

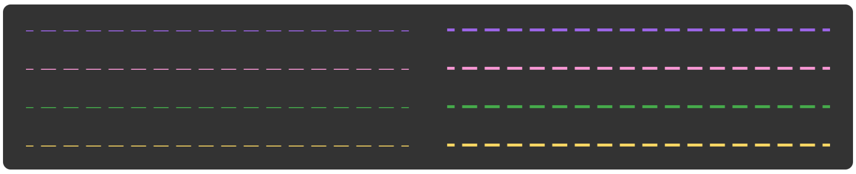


Figure 7.25: High-fidelity iterations for Line Assist. Where color choices and line width were investigated.

During the high-fidelity prototyping of Two Pointy, the color and width parameters were again explored. Different variations of these parameters were tested, however, it resulted in the decision to continue with the current design, making no further iterations.

The Sequence Process Wizard required the most design work to achieve its high-fidelity prototype. Different solutions on how to visualize the various states of the wizard were investigated. It all resulted in indicating the completed steps as filled blue circles with a checkmark icon, the current step as circles outlined with a blue stroke, and the upcoming steps outlined with a white stroke, see Figure 7.26. Additional iterations regarding how to incorporate sub-processes into the wizard were explored. The initial prototype indicated this with a diamond-shaped icon, which was changed to a cuboid icon, see Figure 7.26. This was done since the sub-processes in the wizard involved cuboid editing, making this icon the more logical choice due to its resemblance. Finally, iterations regarding the sizing of the Sequence Process Wizard were investigated, with the decision to make it as small as possible without losing usability. This was done based on the research about UX in 3D platforms, see Section 3.6, and its principle of maximization of workspace.

Figure 7.26: High-fidelity prototyping of the Sequence Process Wizard. Where icons and states were investigated.

The Adjustment Counter and Adjustment Scoreboard were also developed into high-fidelity prototypes. The Adjustment Counter prototype consisted of small iterations, mainly changing font size, style, and spacing to follow Kognic's design system. The Adjustment Scoreboard prototype underwent some bigger iterations, mainly regarding how to indicate a new high score. In the initial prototype, this was indicated with a notification, while the high-fidelity prototype shows this by flashing the icon for the score. This was done since receiving a notification each time a new high score was achieved would become a nuisance, so a more subtle method was necessary. Additional iteration regarding icon options for the scoreboard was also decided upon, see Figure 7.27.

Figure 7.27: High-fidelity iteration of the Adjustment Scoreboard. Where various icon choices were investigated.

The Adjustment Counter and Adjustment Scoreboard are two interesting ideas to help give feedback to the annotators. However, due to time limitations, it was decided that these concepts were not to be evaluated. Instead, they will be concepts for future work, see Section 10.6.

7.3.4 Evaluation

The high-fidelity prototypes of Line Assist, Two Pointy, and Sequence Process Wizard were decided to continue with and needed to be assessed through various evaluation methods, see Section 4.12. This is an important step in the design process to understand what works, what does not, and what improvements could be made during the next iteration phase. To accomplish this, A/B testing, user experience survey, cognitive walkthrough, and design review were utilized, with their processes and results explained below.

A/B Testing

A/B testing was a method used during the evaluation since it gives quantitative data, clearly showing if the new design is successful or not, see Section 4.12.1. Line Assist was A/B tested by 11 annotators and was chosen since it was already developed during the initial development. The same annotators testing Line Assist were also used to gather data from the original design, acting as a benchmark for the test. During both tests, the annotation tasks were sequence annotations of similar nature so they did not vary in difficulty.

The data gathered during the tests were the number of adjustments made per cuboid. In total, 409 cuboids were edited using Line Assist and 1839 cuboids were edited using the original design. The data analysis showed positive results which can be read in Section 8.

User Experience Survey

To get a better understanding of the annotator's opinions about Line Assist, a user experience survey, see Section 4.12.3, was sent out to the participants of the A/B testing. This method was utilized to gather data about their experience with the new tool, specifically focusing on their subjective thoughts. This additional perspective complements the quantitative statistics gained from the A/B testing method.

The survey began with an introduction text explaining its purpose, stating that it is anonymous, and asking for their honest opinions. Following that, the annotators were asked how long they had been using the Kognic platform. The core of the survey was one question about the overall experience, one question on how easy it was to align the cuboid to the ground, and four in-depth questions about their experience using the tool.

The survey consisted of multiple choice questions, balancing scales, and a final text input where the annotators could add additional information that might have been missed. The questions about the overall experience and how easy it was to align the cuboid had a seven-point scale rating system, ranging from awful to great with a neutral center point. The four in-depth experience questions had a five-point scale rating system, labeled awful, bad, neutral, good, and great.

The survey was answered by nine of the annotators, with the majority of them working in the Kognic platform for over one year (n=6). The general results from the survey were positive, with no answer from the scales below neutral. None of the annotators added any additional comments. The entire results can be found in Appendix A.5.

Cognitive Walkthrough

The cognitive walkthrough evaluation method, see Section 4.12.2, was utilized for the Sequence Process Wizard. This method was chosen since it is suitable to evaluate the usability of an interface in a quick and cost-efficient way without a functioning prototype or the involvement of users. The development of the concept would need substantial effort and therefore not be feasible to carry out during this Master's thesis.

The full cognitive walkthrough can be seen in Appendix A.6, the most important and valuable results being:

- The user would not know how to move on to the next step in the wizard. This is done by pressing the tab keyboard button.
- The user might only think they should trim the object visibility spam from the left since that is the icon used for this stage. This could lead them to move on to the next step without finishing trimming the visibility on the right.
- The user might not understand they have finished all keyframes since the wizard tells them to move to the next frame.

- The users might have difficulty understanding that the keyframe annotation is a looping process.

Design Review

The design review was a method used to evaluate the design aspects of all three concepts. It was a 60-minute session where four of the designers at Kognic and the two Master's thesis students participated. Two Pointy and Line Assist were discussed for 10 minutes each since they are simpler concepts with few parameters, while the Sequence Process Wizard was given 30 minutes due to its complexity. The remaining time was used as a buffer time for the introduction of the purpose of the session, finalizing the session, and if discussions would run longer for any of the concepts.

The design review included discussions around parameters such as colors, sizing, icons, and various types of visualizations, in addition to overall thoughts and feedback on the concepts.

Line Assist had in general quite little feedback, the only noteworthy comment was that the dashed line was quite thin and difficult to see. Similar to Line Assist, Two Pointy also did not receive many feedback points. The important aspects were that holding down the mouse button and dragging out the line could become tiring and potentially pose an ergonomic issue for the annotators. This is because they annotate hundreds of objects per day so tools requiring extra finesse and maneuvering become tiring over time. Additionally, feedback regarding whether the line should stay in the object window after using Two Pointy was discussed. The remaining line would have two grab points on its extremities allowing for changes to the rotation of the cuboid after the initial rotation has been completed. This could be useful both if the initial rotation was incorrect, but also for sequence annotation when adjusting the rotation of the cuboid for the remaining frames.

The Sequence Process Wizard was, as previously mentioned, the concepts reviewed the longest. This was because the concept includes several different parameters and would require additional time for discussion. The review resulted in numerous feedback points which are presented in the list below:

- The annotators might not understand what each step in the process wizard is supposed to represent. Additional text would be needed to clarify this.
- The process wizard could potentially block some aspects during annotation, take up screen real estate, and become an irritating factor once the annotation process has been learned since it is no longer needed.
- The white color used for upcoming states is incorrect. Instead, a gray color should be used following Kognic's design system.
- If there is no option to go back to a previous step in the process wizard to correct a mistake, the annotators might instead just delete the object and start from the beginning.

- When a step in the process wizard is completed, the annotator could get rewarded through some type of gamification.
- The final step in the process wizard, reviewing each frame in the sequence, almost defeats the wizard's entire purpose. If each frame is checked and adjusted at the end, the annotators could just use their normal workflow, annotating each frame in chronological order.
- The expandable process wizard (within the original wizard) is not clear enough. The user would not understand what the icon means or what happens when that step is reached.
- The text size indicating the number of keyframes is too small making it difficult to read.

7.3.5 Final Iterations

In this section, the final iteration of the concepts are presented. The final iterations were based on the knowledge gathered from the evaluation of the concepts, see Section 7.2.2. The outcome of these iterations is the final result of this Master's thesis, which can be read about in Section 9. Below, the iterations made to Line Assist, Two Pointy, and the Sequence Process Wizard are described.

Line Assist

In general, Line Assist did not receive much feedback during the evaluation of the concept. There is simply not much to give feedback on since the concept is simple with few design parameters. However, one aspect brought forward during the design review was that the dashed lines extruding from the cuboid are very thin and difficult to see. Currently, annotators can change the width of the cuboid lines in the top menu of the Kognic Platform. The width of the dashed will be changed the same way, through another setting, solving the problem of being difficult to see.

Two Pointy

Two Pointy received more feedback than Line Assist, but the concept is still quite simple with few design parameters. Two feedback points came from the design review, where it was mentioned that the current method of drawing the line could become tiring over time and that the line should maybe remain in the window after it has been drawn.

The first was iterated by changing the concept to include clicking the mouse button once, then dragging out the line, and confirming by clicking the mouse button again. Previously, the concept only included holding down the mouse button while dragging out the line and then releasing it, but the new iteration allows for both. The second feedback point from the design review was decided to not be included in the final design. Having a line with grab points would have resulted in a new rotation tool in the toolbar where the annotators could pick between either the original or Two Pointy. It was decided that a line with grab points on its extremities would most likely not decrease the number of micro-adjustments, it would simply be another way to rotate the cuboid. The powerful feature of Two Pointy is its shortcut command, allowing one quick rotation of the cuboid with no need for additional adjustments. According to the research on designing for sovereign posture, see Section 3.5, expert users appreciate shortcuts, which further reinforce the decision.

Sequence Process Wizard

The Sequence Process Wizard is a complex concept with several design parameters. Therefore, the concept received multiple feedback points from both the cognitive walkthrough and the design review. One of these was the difficulty of understanding the expandable phases within the wizard. To create a better distinction between the expandable phases and the initial wizard, titles were added in the top left corner. The titles used were *Process Guide* for the main phase of the wizard, *Edit Cuboid* for the first expandable phase, and *Edit Keyframes* for the second expandable phase.

Additionally, the icon representing an expandable phase was once again iterated and redesigned, see Figure 7.28. This iteration helped make the icon more understandable since the initial one was already used within the Kognic Platform, making it hard to interpret its meaning. The new icon is built up using three small circles, indicating multiple hidden steps inside it. Finally, to further enhance the understanding of the expandible phases, animations were added to visualize its expansion.

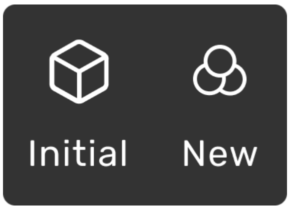


Figure 7.28: The final iteration of the Sequence Process Wizard. Where the expansion icon was once again changed.

Another feedback point was that annotators might not understand what each step in the Sequence Process Wizard is supposed to represent. This could lead to annotators not knowing what to do in each step. In addition to the icons, text describing each step was added in the form of tooltips. The tooltips appear when the annotator hovers over one of the steps in the wizard, explaining it and the procedure for completing it. The cognitive walkthrough showed that annotators would potentially not understand how to move to the next step in the wizard. Therefore, a larger tooltip dialog will appear when hovering over the Sequence Process Wizard itself. This explains the overall concept, including a gif of the wizard, and that the tab key is used to proceed to the next step while pressing shift and tab keys take one back to the previous step. The option to go back to the previous step was another feedback point from the evaluation of the concept. If this action were to be blocked, the annotator might just delete the entire cuboid and restart the process to fix a mistake made in a previous step.

The cognitive walkthrough also showed that annotators might have difficulty understanding that the Edit Keyframes phase is a circular process. Additionally, this would lead to them having a hard time understanding when the phase is completed. To enhance the understanding, the Sequence Process Wizard was redesigned by adding a keyframe counter in the top right corner of the wizard. The counter presents how many keyframes have been annotated out of the total number of keyframes. Further, a dashed line was added between the next keyframe step and the last expansion icon. When all but the final keyframes are completed the line becomes solid and the next keyframe step disappears. All changes made to the Keyframe phase can be seen in Figure 7.29.

Figure 7.29: The final iteration of the Sequence Process Wizard. Showing the initial and new design.

Another feedback point from the design review was that the final step in the Sequence Process Wizard, reviewing each frame in the sequence, almost defeats the wizard's entire purpose. If annotators were to review each frame in the end, they could just use their normal workflow. This was iterated by changing the purpose of the final step. The new purpose of the final step is to set the properties for the remaining frames of the object for each frame. If they see the cuboid be out of place in any of the frames they can fix it, but since it is no longer the purpose of the step, it should in theory nudge them to not make micro-adjustments to those of already sufficient quality. With the new final step, the icon also needed to be iterated which can be seen in Figure 7.30.

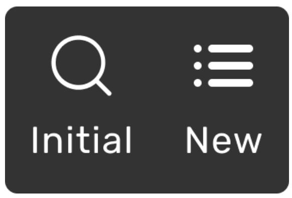


Figure 7.30: The final iteration of the Sequence Process Wizard. Where the icon in the final step was changed.

Additionally, some minor iterations were also made to the Sequence Process Wizard based on feedback from the evaluation. This included changing the white color for the upcoming steps to a gray color, following Kognic's design system, changing font sizes to make it more legible, and making it possible to change the location of the wizard through a grab point.

All iterations made to the Sequence Process Wizard improve its user experience and align it with the Kognic Platform. It adds the final touches to the concept and makes it into its final design.

8

Data Analysis

As mentioned in the process and execution, the A/B testing of Line Assist showed positive results, see Section 7.3.4. The data was analyzed by comparing the number of adjustments made per cuboid using Line Assist to using the original design. Additionally, each cuboid editing tool was independently analyzed to the benchmark. In total, 11 annotators edited 409 cuboids using Line Assist and 1839 cuboids using the original design. In each of the graphs below, the blue bars represent using Line Assist, the orange bars represent using the original design, and the brown sections are the overlap between the two. The Translate Mode tool showed a reduction of 0.31 adjustments per cuboid, where the original design had an average of 3.75 adjustments and Line Assist had an average of 3.44 adjustments, see Figure 8.1.

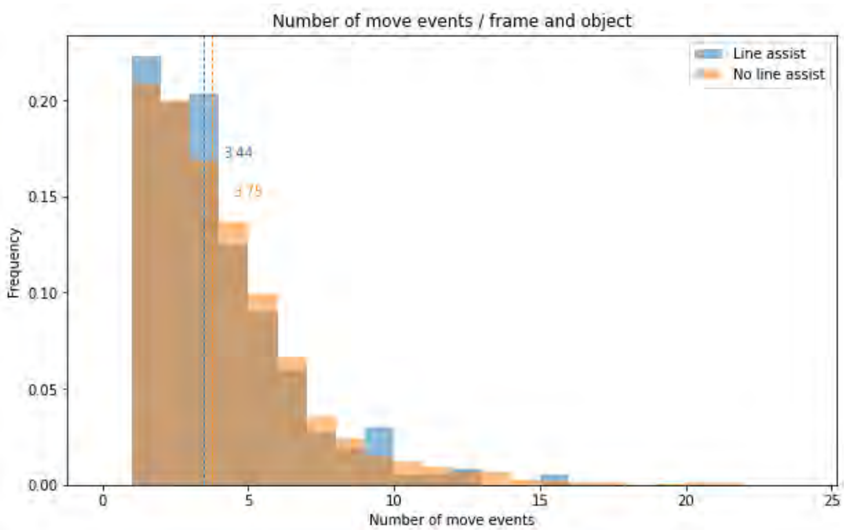


Figure 8.1: Data analysis for the Translate Mode tool, comparing Line Assist to the original design.

The Rotate Mode tool showed a reduction of 0.41 adjustments per cuboid, where the benchmark had an average of 3.53 adjustments and Line Assist had an average of 3.12 adjustments, see Figure 8.2.

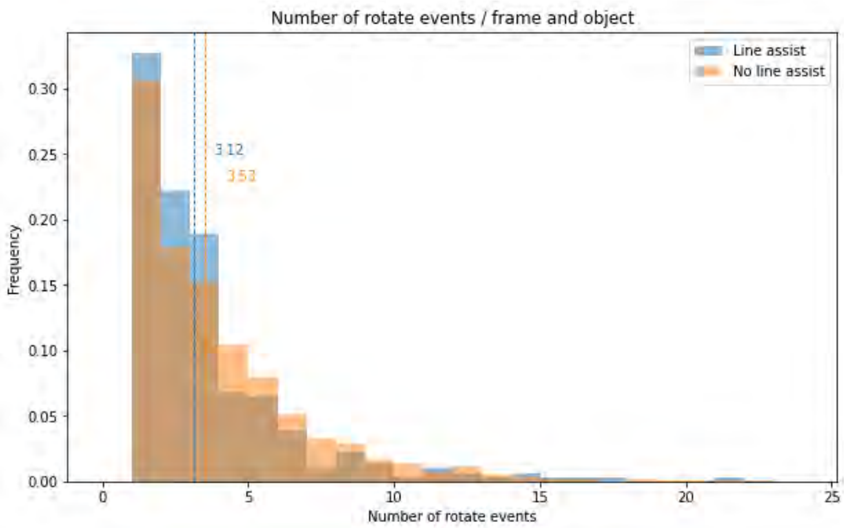


Figure 8.2: Data analysis for the Rotate Mode tool, comparing Line Assist to the original design.

The Scale Mode tool showed a reduction of 0.49 adjustments per cuboid, where the original design had an average of 2.41 adjustments and Line Assist had an average of 1.92 adjustments, see Figure 8.3.

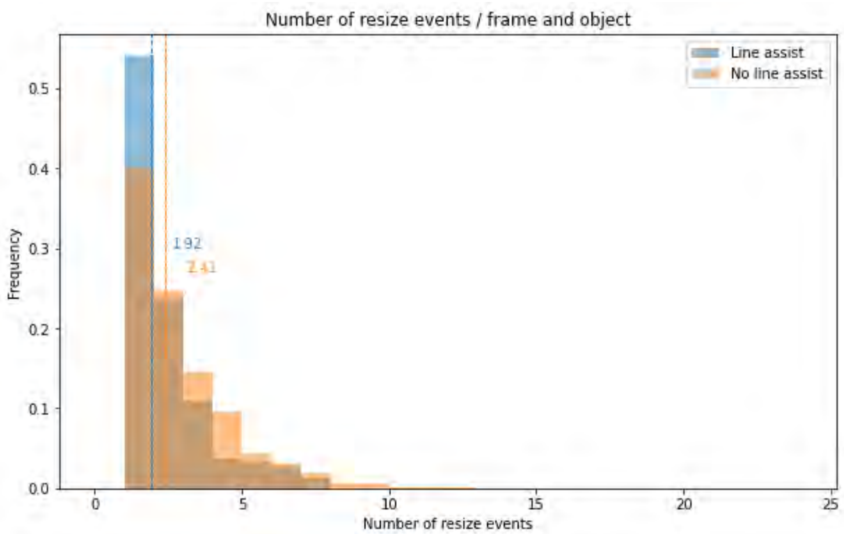


Figure 8.3: Data analysis for the Scale Mode tool, comparing Line Assist to the original design.

When analyzing the overall editing events, there is a reduction of 0.87 adjustments per cuboid, where the benchmark had an average of 7.34 adjustments and Line Assist had an average of 6.47 adjustments, see Figure 8.4.

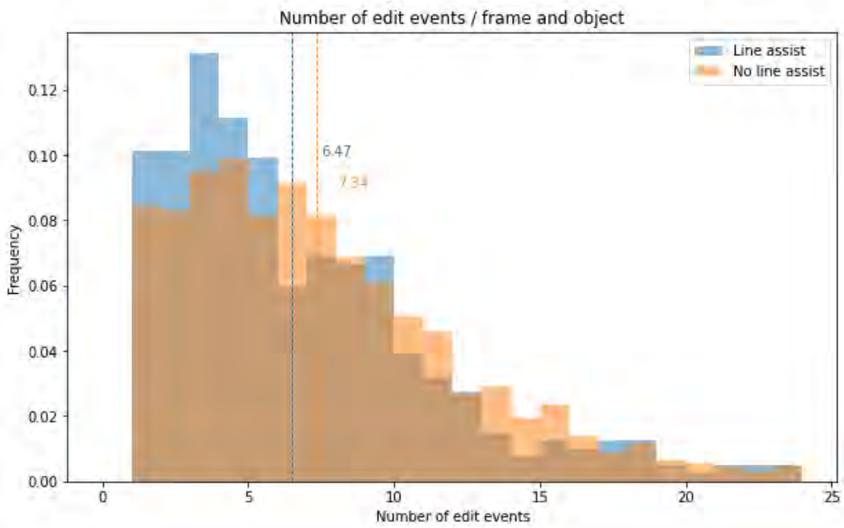


Figure 8.4: Data analysis for the total editing events, comparing Line Assist to the original design.

A reduction of 0.87 adjustments per cuboid might appear as little, but it is a reduction of 12% from the original design. Additionally, if Line Assist had been used for the 1839 cuboids edited without it, it would have saved approximately 1600 adjustments. Discussion regarding the reliability of the results, the quality of annotations, why a reduction in adjustments occurs during translate and scale events, and why the overall number of adjustments is lower than the previously calculated optimal number of adjustments can be read in Section 10.2.2.

9

Result

This chapter presents the results of this Master's thesis through the final design solutions and the design guidelines. Furthermore, the research question will be answered based on the contents of the thesis.

9.1 Final Design Solutions

In this section, the final design solutions, Line Assist, Two Pointy, and the Sequence Process Wizard will be explained in detail. Together, the final designs fulfill most of the guidelines and requirements specification, each focusing on different areas, which is further discussed in Section 10. However, they all independently accomplish the must have categories. These design solutions are the results of the process and execution of the Master's thesis, see Section 7.

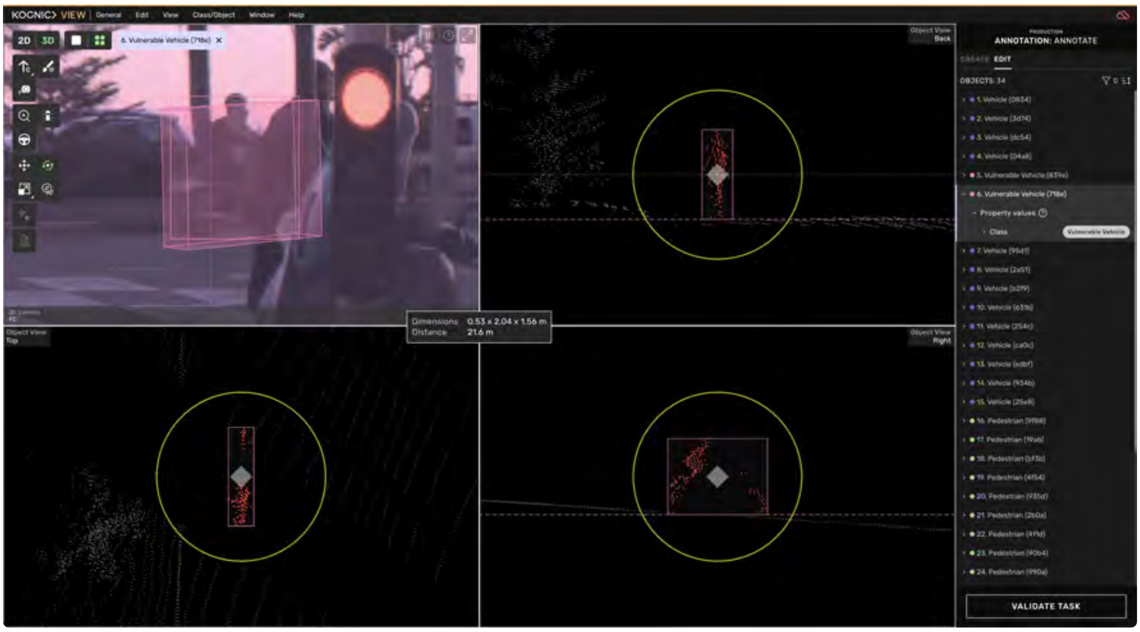


Figure 9.1: Overview of Line Assist.

9.1.1 Line Assist

Line Assist is a tool to help annotators align the cuboids horizontally to the ground by providing assisting lines. These are dotted lines extruding from the base of the cuboid, allowing the annotators to utilize the surroundings when adjusting the rotation. The width and color of the dotted lines are directly correlated to the lines of the cuboid, which means they vary depending on the used settings and the class of the object. The dotted lines are always visible in the back and top object view, and will appear once the annotator enters the Grid View, see Figure 9.1.

9.1.2 Two Pointy

Two Pointy is a tool that helps annotators align the cuboid horizontally to the ground and works by dragging out a line to rotate the cuboid, see Figure 9.2. The tool can be accessed and used anytime while in Grid View in the back and right view object windows.

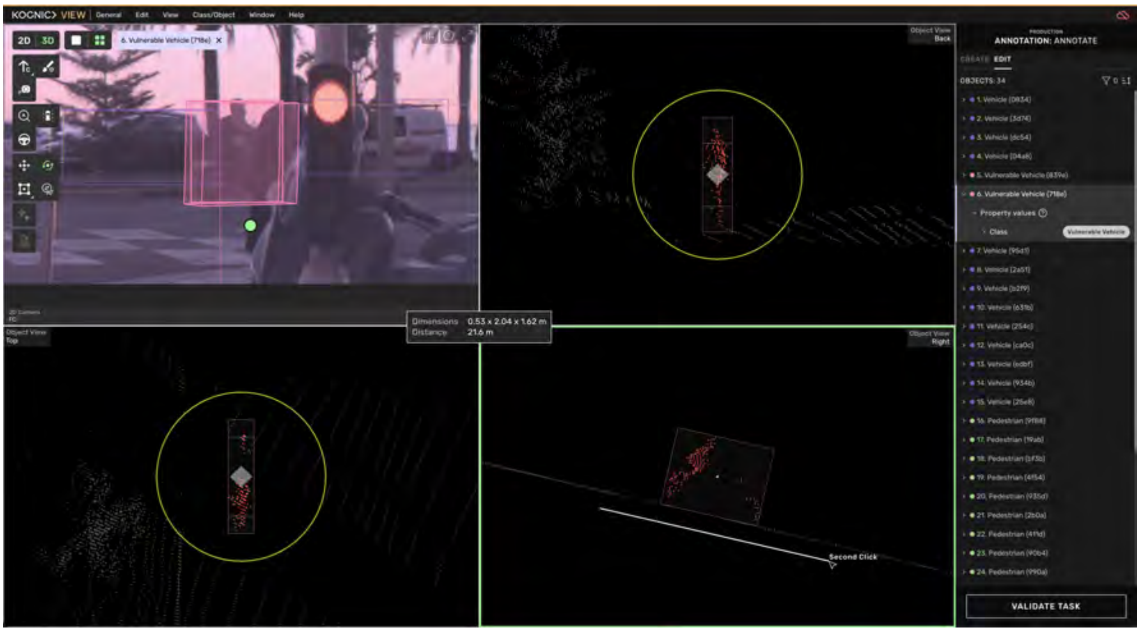


Figure 9.2: Overview of Two Pointy

Two Pointy is a shortcut tool accessed by holding down the shift keyboard button. This allows the annotator to place out one point by pressing the mouse button and drag out a line by moving the mouse, see Figures 9.3 and 9.4. The line is always parallel to the cuboid, so by changing the angle of the line the annotator can rotate the cuboid. To finish the rotation, the second point is placed, again by pressing the mouse button, see Figure 9.5. The line can also be drawn by pressing and holding down the mouse button, moving the mouse, and releasing the mouse button to finish the rotation.

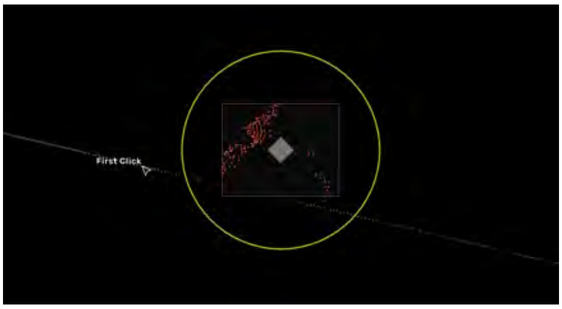


Figure 9.3: Two Pointy Step 1.

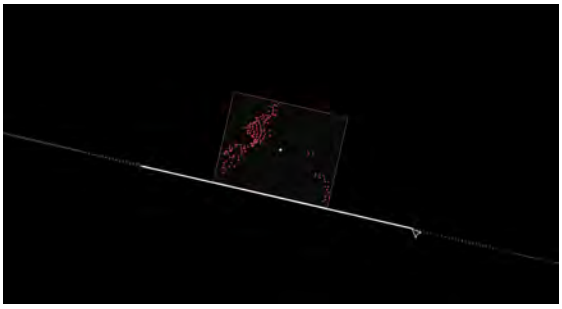


Figure 9.4: Two Pointy Step 2.

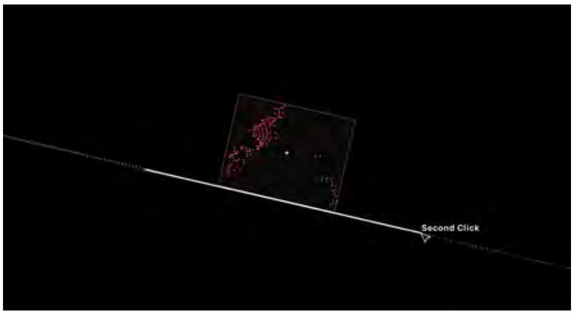


Figure 9.5: Two Pointy Step 3.

As seen in the images, the ground is utilized to draw the line, perfectly aligning the cuboid horizontally to the ground. This is the recommended use, however, the line can be drawn anywhere on the screen, for example if the ground is not visible. Additionally, the width of the assisting lines can be adjusted in the top menu through a setting.

Figure 9.6: Overview of the Sequence Process Wizard.

9.1.3 Sequence Process Wizard

The Sequence Process Wizard guides annotators through an optimal and efficient workflow for sequence annotation. Its purpose is to reduce the number of micro-adjustments by indicating and assisting users with what tools to use and when. The wizard is located at the bottom of the screen, above the sequence video bar, in the center of the main working window, see Figure 9.6. This position minimizes its invasiveness by not blocking any necessary functions or screen real estate.

The Sequence Process Wizard has three phases, one main phase called Process Guide and two expanding phases called Edit Cuboid and Edit Keyframes, see Figures 9.7, 9.8, and 9.9. This allows the wizard to include all aspects of sequence annotation while keeping a small stature.

Figure 9.7: Process Guide Main Phase.

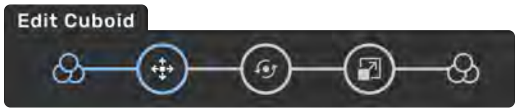


Figure 9.8: Edit Cuboid Expanding Phase.

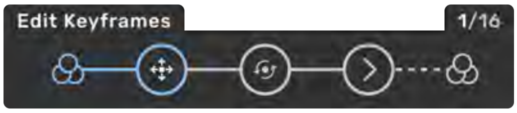


Figure 9.9: Edit Keyframes Expanding Phase.

Below the order of operations of the Sequence Process Wizard will be explained. The first step in the wizard begins in the Process Guide main phase.

1. Center Mode tool to draw out the cuboid in the frame where the object is most visible in the Main View, the 3D point cloud.
2. Opens the Edit Cuboid expansion phase and automatically switches to Grid View.
   (a) Translate Mode tool to move the cuboid into the approximate correct location.
   (b) Rotate Mode tool to rotate the cuboid, aligning it horizontally to the ground and to the object itself.
   (c) Scale Mode tool to transform the cuboid into the correct dimensions.
   (d) Close the Edit Cuboid expansion phase.
3. Trim the object's visibility. This is done by marking out beginning and end-points, in addition to setting the object to not visible if necessary in any of the frames.
4. Opens up the Edit Keyframes expansion phase and automatically switches to Main View.

(a) Translate Mode tool to move the cuboid into the correct position.

(b) Rotate Mode Tool to rotate the cuboid into position.

(c) Right arrow keyboard button to move to the next keyframe.

(d) Close the Edit Keyframes expansion phase.

5. Set the properties of the object for the remaining frames.

During the expansion and contraction of the Edit Cuboid and Edit Keyframes, the Sequence Process Wizard utilized animation. This visualizes the opening and closing of the phases, allowing the annotators to understand what is happening more easily. During the Edit Keyframes expansion phase, the keyframes are automatically set to every five frames in the sequence, locking the others. This is a circular process where the annotator goes through the wizard several times for each keyframe. During the final step in the Process Guide main phase, if the annotators see the cuboid be out of place in any of the frames they can fix it. However, since it is not the purpose of the step, it should nudge them not to make micro-adjustments to those of already sufficient quality.

In the top right corner of the Edit Keyframe, a counter presents how many keyframes have been annotated out of the total number of keyframes. When all but the final keyframe is annotated, the dashed line becomes solid and the next keyframe step will disappear, see Figure 9.10.

Figure 9.10: Edit Keyframes Expanding Phase for the final keyframe.

To indicate completed, active, and upcoming steps, the Sequence Process Wizard utilizes various states, see Figure 9.11. This should also nudge the annotator not to go back to a previous step and make micro-adjustments since it has already been completed.

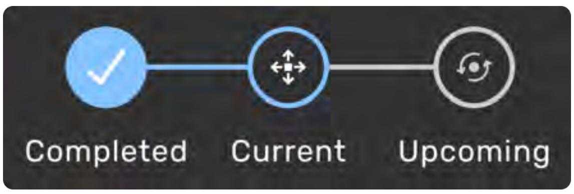


Figure 9.11: The various states in the Sequence Process Wizard.

The Sequence Process Wizard utilizes automatic progressing when possible, but for the majority of steps in the wizard, the annotator needs to advance to the next step themselves. This is done by pressing the tab keyboard button. To go to the previous step, the annotator can hold down the shift keyboard button and then press the tab keyboard button. This, and the Sequence Process Wizard, is explained through a larger tooltip dialog, see Figure 9.12. The tooltip appears while hovering with the mouse over the wizard.

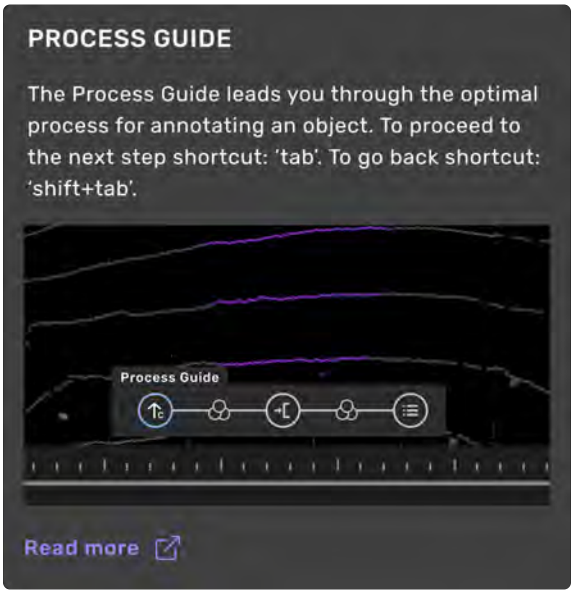


Figure 9.12: The larger tooltip dialog for the Sequence Process Wizard.

Additional tooltips are also provided to the annotator while hovering with the mouse over one of the steps in the wizard. The various tooltips are listed below and an example of its appearance is shown in Figure 9.13.

- Center Mode Step - *Use Center Mode in the frame where the object is most visible*
- Trim Object Visibility Step - *Trim object visibility*
- Properties Step - *Set object properties for remaining frames*
- Edit Cuboid Expansion Phase - *Expandable phase for editing the cuboid*
- Edit Keyframe Expansion Phase - *Expandable phase for keyframe annotation*
- Translate Mode Step - *Use Translate Mode to translate the cuboid*
- Rotate Mode Step - *Use Rotate Mode to rotate the cuboid*
- Scale Mode Step - *Use Scale Mode to scale the cuboid*
- Next Keyframe Step - *Use the Right Arrow Key to move to the next keyframe*

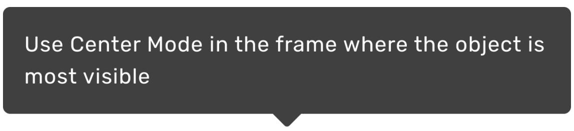


Figure 9.13: Example of the tooltip for each step in the Sequence Process Wizard.

The final feature of the Sequence Process Wizard is that the annotator can move the wizard to any desired position on the screen. This allows for customizability and is done by using the grab handle on the left side of the wizard.

9.2 Rejected Design Solutions

This section will present the two rejected design solutions, Adjustment Scoreboard and Adjustment Leaderboard, created during the high-fidelity prototyping, see Section 7.3.3. Even though the concepts were rejected due to time limitations, they have potential for future work, see Section 10.6.3.

9.2.1 Adjustment Scoreboard

The Adjustment Scoreboard allows the annotator to keep track of the lowest number of adjustments they made while annotating an object, see Figure 9.14. This is done by keeping track of the top three lowest number of adjustments executed to annotate an object within the current task. These are then presented to the annotator through a scoreboard and aim to reduce micro-adjustments by gamifying the task. The scoreboard is individual, meaning that only the users can see their adjustments, reducing the risk of creating a competitive environment between annotators. It includes three medallions, correlating to the number of adjustments: gold, silver, and bronze. Once a new high score, or in this case, a low score, is reached, the medallion and number flash to indicate this. To enhance the understanding of the scoreboard, a tooltip was added that appears while hovering with the mouse. The tooltip explains the Adjustment Scoreboard as follows: *Scoreboard for least adjustments per cuboid*.

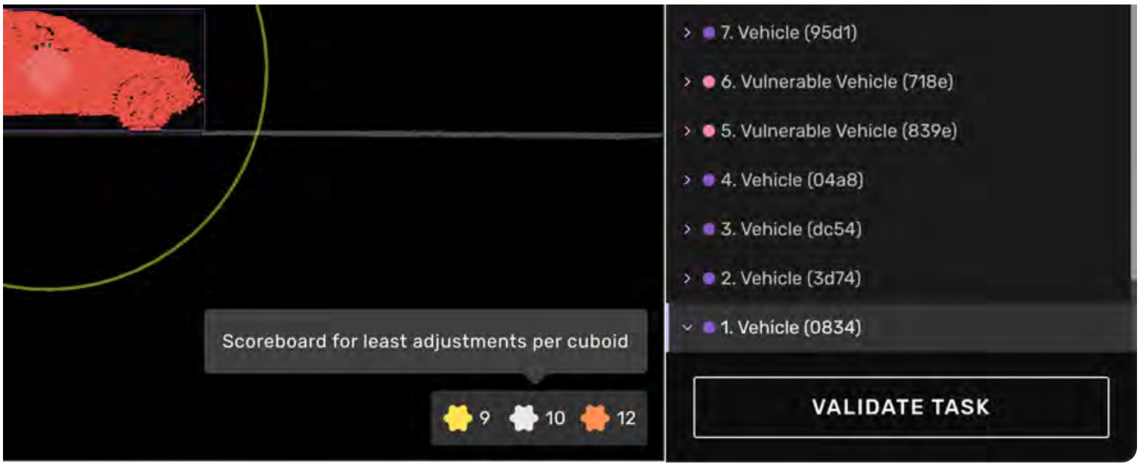


Figure 9.14: Adjustment Scoreboard final design.

9.2.2 Adjustment Counter

The Adjustment Counter aims to help annotators reduce the number of micro-adjustments by providing feedback. In the top right corner, beside the timer that indicates the amount of time spent on the currently selected object, an adjustment counter was introduced, see Figure 9.15. This counter indicates the number of adjustments made on the currently selected object, as adjustments are made, the counter increases. Additionally, to enhance the understanding of the concept, a tooltip was added which opens upon hovering the counter with the mouse. The tooltip explains the adjustment counter as follows: *Total number of adjustments made to currently selected object*

Figure 9.15: The Adjustment Counter and corresponding tooltip.

To further give feedback to the annotators, a status bar was introduced in the bottom left corner of the platform, see Figure 9.16. This supports the adjustment counter and the timer by providing information about targets and averages for the session. It allows for continuous feedback to the annotators, displaying the target time and target numbers of adjustments per object in addition to the average time spent and average number of adjustments per object.

Figure 9.16: The status bar for the Adjustment Counter.

9.3 Design Guidelines

In this section, guidelines regarding what to consider when designing to reduce micro-adjustments during annotation will be presented. The design guidelines were created with the learnings from this Master's thesis as a base, creating a structure for an approach to similar problems in the future.

1. Follow a Design Process When designing to reduce micro-adjustments during annotation, following a design process is essential. There are multiple design processes to choose from depending on the project and personal preferences. The most important aspects are involving the users and stakeholders throughout the process and allowing for iterations. Utilizing a design process creates a suitable structure for approaching the design problem, including all necessary aspects to consider.

2. Incorporate Design Theories Designing for a 3D point cloud annotation process is still quite a novel field with few previously documented cases and projects. Since the topic has yet to be vastly explored, relying on already existing and proven design theories is important. This allows for a solid structure that later can be used as a basis and further elaborated. Carrying out a literature review is therefore an essential step in setting up a successful final design.

3. Gather Information from Multiple Sources Business process outsourcing (BPO) provides annotation services and access to their annotators is a challenging task. The BPOs are likely located abroad, hence, different information-gathering methods need to be utilized. For user research with annotators, online meetings for interviews and video-recorded observations are recommended. Due to their location, it can be difficult to obtain a large pool of annotators and gather a sufficient amount of information. To combat this, acquire information from multiple sources, for example conducting a workshop with knowledgeable employees at the company.

4. Consider Annotator's Culture The annotators' culture is naturally embedded into their work, an important topic to consider when designing for reducing micro-adjustments. This might prompt them to aspire to create flawless annotations, influenced by the culture of perfectionism. With the correct guidance, tools, and feedback, unnecessary work, such as micro-adjustments, can be reduced.

5. Align Stakeholders Annotators need help understanding when a cuboid has reached sufficient quality. This needs to be considered when approaching the design problem. To accomplish this, it is crucial to align stakeholders and have structured information for the annotators to follow. The customer, together with the company, needs to specify precise criteria of what is considered sufficient quality.

6. Consider all Areas of Concern Micro-adjustments occur in different parts of the annotation process so all areas of concern must be considered. This Master's thesis was limited to 3D object annotation of cuboids, however, it was discovered that micro-adjustments also appear during both 2D annotation and the review phase. It is therefore important to either focus on one specific problem area or understand the entire annotation process.

7. Micro-Solutions to Minimize Micro-Adjustments Designs created to reduce micro-adjustments do not have to be rigorous, resulting in a complete change of the interface or the annotation process. Several small design solutions, such as providing feedback, changing interactions, or providing guidance can efficiently change the way annotators perceive their work, leading to a reduction of micro-adjustments.

8. Improve Existing Tools Over Creating New Ones Creating new tools to help reduce the number of micro-adjustments is not always the best solution, tools that exist could instead be optimized. Therefore, begin with redesigning tools that already exist. This could drastically reduce the number of micro-adjustments without introducing a new tool annotators have to learn how to use. If improvements are not possible or they do not help solve the problem of micro-adjustments, then one can consider the design and implementation of new tools.

9. Provide Clear and Efficient Feedback Annotators need feedback from the platform to understand when an annotation has reached sufficient quality, the lack of it will provide inadequate guidance, potentially leading to micro-adjustments. Hence, it is important to utilize the criteria gathered by aligning all stakeholders. These criteria can be converted into guidelines or design aspects that are easy for annotators to interpret and follow. Feedback to annotators needs to be introduced in the design wherever it is possible to reduce the number of micro-adjustments.

10. Use Automation when Possible To guide the annotators through the annotation process, automation should be introduced when possible. This can allow for the reduction of micro-adjustments by helping annotators with their work. However, one must consider that annotators still need to feel like they are contributing.

11. Focus on Rotation Micro-Adjustments During this Master's thesis, rotation micro-adjustments were especially prominent. This area of concern should be of interest while designing for the reduction of micro-adjustments. However, important to consider is that this might not be the case for all annotation platforms. Each platform could have its own problem areas, hence the importance of following the first design guideline, and using a design process for approaching the design problem.

10
Discussion

In this chapter, various discussion points regarding the process and execution, result, generalizability, ethical issues, and future work are presented.

10.1 Reflection on Process and Execution

This section presents discussions of some parts of the process and execution of this Master's thesis. It will analyze the change of scope, the limited contact with end users, and the already existing solutions.

10.1.1 Change of Scope

The transition from solely focusing on single-frame annotations to including sequence annotations was made early in this project. This decision was reached after consulting our advisors at Kognic, who suggested the change of scope. However, the change in scope occurred after the initial collection of data had begun, resulting in the observation videos capturing only single-frame annotations. The knowledge regarding the sequence annotation process was instead gathered from annotating ourselves, the explanation given by the expert annotators during the interviews, and during the workshop with Kognic.

While it is conceivable that the change could have influenced our result, given our initial exposure to single-frame annotation, the exact impact is uncertain. An effect it potentially could have had is that our focus leaned more toward single-frame, slightly neglecting processes only applicable to sequence annotation. However, even if that were the case, it would not have significantly impacted the result. There are substantial similarities between the two types of tasks, with some processes even being identical. Solutions produced for single-frame annotations would, therefore, benefit sequence annotation as well. One problem would have arisen if designs only applicable to sequence annotations had not been explored at all, but that was not the case.

10.1.2 User Research

One limitation throughout the whole project was the limited direct communication with the annotators. Since the annotators are located outside of Sweden and have limited proficiency in English, we had few opportunities to talk to them. Additionally, since Kognic offers various annotations to their customers, their users are not solely focusing on point cloud annotations, they have other ongoing projects as well. This further constrained the potential research population. Due to this limitation, we were only able to recruit a small number of participants during our user research, especially during our observations and interviews, while our workshop included a sufficient number of participants. While these sessions provided valuable insights, complementing each other, it is evident that a larger research population could have reached a more reliable result.

The observation conducted felt comprehensive and provided valuable insights into the annotators' work process. Nevertheless, they did not yield any new insights by the end, indicating that the data collection reached a point of saturation where further accumulation of information would not lead to additional insights. Similarly, the workshops gathered plenty of insights from key individuals within the company, resulting in a comprehensive data collection. However, the interviews with the expert annotators presented a potential opportunity for further exploration, as some new aspects arose during the second interview.

The diverse workshop group offered valuable perspectives on the problem area, complementing the limited direct contact with the annotators. While it is important to acknowledge the constraint of having limited interactions with the annotators, we would argue that involving knowledgeable people at Kognic enhanced the result's reliability. Another important aspect is that it enabled us to overcome the problem of collecting insufficient data. Additionally, it provided insights that would not be possible to achieve only by approaching the end users. This included information regarding previous work already tested by Kognic, as well as the limitations and possibilities for the design to rely on automation.

How a larger number of interviews with expert annotators would have affected the end result is hard to tell. It could potentially have given a broader perspective of how different annotators perceive the different tools and approach annotation tasks. Whether that additional knowledge would have a significant impact on the final result is uncertain. However, it is likely that it may not have made a substantial difference, considering the comprehensive data collected from the observations and the workshop.

10.1.3 Previously Tested Concepts

During the concept evaluation, see Section 7.2.2, the early stages of some concepts were presented to a developer at Kognic. This meeting resulted in learning that some of the ideas had already been tested previously and proven to be inefficient. This discovery was particularly interesting, since we, unaffected by each other's works, had developed similar solutions, at different periods. Additionally, it was a valuable asset to the development of the design, since the focus could be shifted to the ideas more likely to succeed, instead of wasting time on concepts proven inefficient.

However, it is worth noting that while learning from previous results can streamline the design process, new perspectives could lead to a new variation of the design, one fulfilling its purpose. Even if this would have been the case, learning from previous knowledge was still helpful during this project. Given its limited timeframe, exploring new areas would be more usable and less prone to failure than trying again on an already-tested concept.

10.1.4 Selecting Concepts for A/B testing

Why was Line Assist A/B tested and not Two Pointy? This is one question that might arise while reading this Master's thesis. Both concepts are excellent candidates for A/B testing and can measure the same condition, the number of cuboid adjustments. Additionally, both concepts had been already developed in the testing environment of the Kognic Platform.

The original plan was therefore to A/B test both concepts, but due to time constraints and difficulty in finding a suitable project with enough participants, only Line Assist was tested. This is because it was an easier concept with less impact on the annotations. Two Pointy directly affects the annotation process and would require a project where mistakes can happen. This took longer than expected to find and set up, which left not enough time to complete the test.

10.2 Reflection on Result

3D point cloud annotation platforms are still a novel field that is yet to be fully explored. Even if this research focuses on a specific part of the annotation process, micro-adjustments, it contributes to the future of the field by showing that interaction design methods are suitable when approaching similar problems.

This section discusses the results of this Master's thesis. It will analyze micro-solutions as final designs, the A/B testing of Line Assist, and the results of the questionnaire.

10.2.1 Micro-Solutions

This Master's thesis resulted in three final design solutions, each satisfying the must have criteria outlined in the requirements specification, see Section 7.1.7. Upon reviewing the should have and could have criteria, each concept addressed a subset of them, collectively fulfilling all requirements. Creating multiple micro-solutions to reduce micro-adjustments was proven to be an efficient approach during this project. This since it made it possible to design solutions addressing multiple areas, rather than attempting to incorporate everything into a single solution. Additionally, it allowed for the possibility to implement the designs without requiring any rigorous development. However, one thing that is important to acknowledge with this approach, is that even if the designs are created individually, they need to be either compatible with each other or to be weighed against each other to decide which one is the superior solution.

While the micro-solutions approach offered many advantages, a larger comprehensive design, and changing the entire work process might have an even greater impact on the reduction of micro-adjustments. However, such a solution would be more complex and challenging to design and implement. It might also require a full redesign of the annotation platform, which simply is not feasible. Whether or not one should approach a similar design problem with multiple small solutions or one large solution depends on the conditions of the project, including factors such as time and resource limitations. During this project, leveraging micro-solutions was therefore the superior approach, since it allowed for creating several fully functional designs in a relatively short period.

10.2.2 A/B Testing Line Assist

The results from the A/B testing of the Line Assist showed a reduction for all three types of adjustments, where the biggest improvement was scaling adjustments by a small margin, see Section 8. This is an interesting result since Line Assist was originally created to reduce rotation adjustments. One reasonable cause of this result is that Line Assist enables annotators to utilize the surroundings when scaling to align the cuboids to the ground, providing more efficient feedback. Another interesting insight from the data analysis is that Line Assist also reduces the number of translation adjustments. The cause for this is less obvious, but most likely it is a reduction when translating and aligning the cuboid to the ground for each of the frames in the sequence. To fully understand exactly how Line Assist facilitates the annotators, observations would be needed.

Since the data was gathered from a sequence annotation task, the results can not be compared to the data presented in Section 7.1.3, due to it consisting of single-frame annotations. This is because the data from the A/B testing includes both cuboids created from scratch and cuboids only adjusted in other frames, which involves fewer adjustments. However, this will not affect the outcome of the A/B testing since both the new design and the current design were tested during the evaluation. Hence, the average amount of adjustments will be significantly lower than during single-frame annotation. It would be interesting to gather and compare data from single-frame tasks as well since it potentially could show even further reduction of adjustments.

One aspect worth mentioning is that no data about the quality of the cuboids during the test was collected. To fully understand the improvements Line Assist provides, it would be necessary to know which effect it has on the quality of the cuboids. This would need to be explored before implementing it on full scale.

10.2.3 User Experience Survey

The results from the survey were almost solely positive, with only two neutral answers. Additionally, no written constructive feedback was provided. Even if this implies that the annotators are positive to Line Assist, it has to be taken into account that it can depend on other aspects as well. When creating the surveys, we asked our mentors at Kognic for advice, where they mentioned that previously they had seen a tendency of annotators always being rather positive during surveys. Hence, the reliability of the result is questionable and should be seen as a guidance rather than a fact. The survey is more of a validation of annotators supporting the concept than an evaluation. It would have been a better result with some constructive criticism, allowing for iterations.

10.3 Generalizability

The generalizability of this Master's thesis is quite low. It would be difficult to apply the results to a broader context. The results are somewhat restricted to the subject of annotation, especially micro-adjustments. However, one aspect of the result that can be applied to a broader context is how to handle a small subject pool of end users located far away from where the project takes place. Utilizing video-recorded observations, online meetings for interviews, and most importantly, gathering information from multiple sources are findings that can be generalized. These methods apply to any project with the constraint of limited access to users.

10.4 Ethical Issues

At the beginning of this Master's thesis, ethical issues regarding fair working conditions within the annotation sector were brought up, see Section 6.1. This section further explains Kognic's actions to ensure this and what the Master's thesis could provide to do the same. The goal was that the reduction of micro-adjustments aids in providing the right tools and processes for annotators. If successful, it could support better work conditions since it will improve the platform's user-friendliness, reducing the manual effort needed during the annotation process.

The question remains if the results accomplish this goal. The Sequence Process Wizard aims to reduce micro-adjustments by guiding the user through annotation. Two Pointy aims to reduce micro-adjustments by providing a more efficient tool for rotating the cuboid. Additionally, the A/B testing of Line Assist showed through quantitative data that the concept reduces micro-adjustments. Claiming that these design solutions actually support better working conditions could lead to a false sense of security, however, one should not dismiss that they potentially could.

10.5 Limitations

In this section, discussions regarding the limitations set during this Master's thesis will be reviewed. It will include whether the annotation task is 2D or 3D and the limitation to how annotators interact with physical tools while annotating in the Kognic Platform.

10.5.1 Is the Annotation Task 2D or 3D?

One limitation established for this Master's thesis was to focus exclusively on 3D point cloud annotation, despite the Kognic Platform consisting of both 2D and 3D annotation tasks. However, examining 3D point cloud annotations reveals that they primarily involve 3D elements but also incorporate 2D aspects. When the annotator uses the grid view, the point cloud is transformed into four 2D views: camera image, top view, back view, and side view. Since the grid view and 3D point cloud are highly integrated, annotators frequently switch between them. As a result, the project mainly focused on it being a 3D task, although it could have been beneficial to look into 2D research as well.

10.5.2 Interaction with Physical Tools

This Master's thesis was limited to only focusing on the Kognic Platform since the annotators are based outside of Sweden. It was not possible to investigate how annotators use physical tools to interact with the platform. Because of this limitation, research within HCI modeling human behaviors such as hand-eye coordination and looking at repetitive behaviors was not explored in this Master's thesis. The use of physical tools and human behaviors could potentially provide a broader understanding of the annotation process and yield further insights into the problem area.

One aspect of interest is that the annotators use a mouse as an input device, therefore it would be interesting to look into concepts such as clutching. Clutching involves the action of temporarily lifting the device and repositioning it to extend the range of motion [96]. Another interesting topic to consider would be other input modalities, for example, the usage of gaze input. Gaze input refers to when a user interacts with an interface through eye tracking [97]. One final interesting topic is the use of virtual reality for annotation tasks. However, this area has more potential, both positively and negatively, and is therefore a topic of future work, see Section, 10.6.4.

10.6 Future Work

In this section, future work will be discussed. It will include potential improvements to the final design solutions, Two Pointy and Sequence Process Wizard, and future work necessary for the rejected design solutions. The section will also discuss the possibility of the use of virtual reality within the annotation tasks.

10.6.1 Two Pointy

Two Pointy was one of the final design solutions for this Master's thesis that needs further evaluation to test its viability. Based on the design review it is an interesting concept with potential for reducing micro-adjustments. Similar to Line Assist, A/B testing should be used to get quantitative data for how the concept compares to the current design. It will result in clear data showing if Two Pointy is either superior or inferior regarding the number of adjustments made per cuboid. Additionally, qualitative data can also be gathered with the same survey used for Line Assist.

10.6.2 Sequence Process Wizard

The Sequence Process Wizard was another of the final design solutions of this Master's thesis. However, there is still some future work that would be beneficial for the concept. The first would be to look into gamification aspects of the concept. This could reward the annotator both throughout the process as each step is completed and at the end of the entire process. Important to consider is that the gamification would need to be subtle since the steps in the process are completed frequently and could become an annoyance.

There is also the potential for the Sequence Process Wizard to extend beyond sequence annotation. It can be applied and adapted whenever there is a process with multiple steps that need to be completed subsequently. One promising area to apply this concept is for single-frame annotation, where the steps in the wizard would need to be iterated. The Sequence Process Wizard could also become a great tool for onboarding new annotators, teaching them the most efficient annotation process.

Finally, the Sequence Process Wizard would require further development and user testing to evaluate the viability of the design solution. From the design review and cognitive walkthrough the concept looks promising, however, it is impossible to tell without performing final evaluations.

10.6.3 Rejected Design Solutions

The Adjustment Scoreboard and Adjustment Leaderboard were rejected concepts due to time limitations. These are interesting concepts with the potential to reduce micro-adjustments, but additional work would be necessary before implementation.

The Adjustment Scoreboard was created to gamify the annotation process, potentially increasing the annotation quantity without reducing the quality, which has been proven to be the case in earlier studies, see Section 3.7. To introduce the Adjustment Scoreboard into the Kognic Platform, development, and user testing would be required to evaluate the feasibility of the concept.

The Adjustment Counter helps give feedback to the annotators, something that is limited in today's design. Before implementation of the concept into the Kognic Platform, it requires a deep dive into the organization and aligning all stakeholders on what the target number of adjustments should be and when an annotation is good enough. Each client requires different levels of accuracy, so there is no one number to use across the board. Kognic, together with its client, would need to investigate that number independently. Additionally, further development and user testing is necessary to evaluate the feasibility of the concept.

10.6.4 Utilizing Virtual Reality for Annotation Tasks

The use of virtual reality for annotation tasks was not explored during this Master's thesis due to the limitations of how annotators interact with physical tools while using the Kognic Platform. However, it is an interesting topic and a field that is still growing where its potential in many fields has yet to been explored. The impact of virtual reality on this project is uncertain and could yield both positive and negative outcomes. Integrating virtual reality into annotation tasks would require significant additional work and could likely warrant an entire study on its own.

11

Conclusion

Autonomous vehicles are a growing subject within the urban mobility sector, potentially revolutionizing the industry. For autonomous vehicles to work as intended, their machine-learning models need to be extensively trained to maneuver in a diverse traffic landscape through annotations. This Master's thesis aimed to reduce micro-adjustments by increasing annotator efficiency through interaction design with the research question:

In which aspects can interaction design support the reduction of micro-adjustments (small repetitive unnecessary adjustments of annotations of already sufficient quality) in annotations of 3D point clouds?

From the execution of this project, interaction design supports the reduction of micro-adjustments from different aspects. The study utilized aspects such as design thinking, user-centered design, and the triple diamond design process to address the wicked problem. More specifically, interaction design supported a broad understanding of how users interact with the software, which was crucial to finding areas prone to micro-adjustments. Through this knowledge, the abundance of rotation-micro adjustments in the platform surfaced, resulting in two design solutions, Line Assist and Two Pointy. Another aspect was user evaluation, allowing for the designs to be tested and iterated, leading to more optimal final design solutions.

The result from the design process is three final design solutions, Line Assist, Two Pointy, and Sequence Process Wizard, and 11 design guidelines:

1. Follow a Design Process
2. Incorporate Design Theories
3. Gather Information from Multiple Sources
4. Consider Annotators Culture
5. Align Stakeholders
6. Consider all Areas of Concern
7. Micro-Solutions to Minimize Micro-Adjustments
8. Improve Existing Tools Over Creating New Ones
9. Provide Clear and Efficient Feedback

10. Use Automation when Possible

11. Focus on Rotation Micro-Adjustments

This Master's thesis showed that the iterative nature of interaction design is an efficient approach when designing to reduce micro-adjustments. Following the design process and utilizing various interaction design methods creates a structured path to answer the design problem. While the design solutions show that micro-adjustments can be reduced, see Section 8, the design guidelines explain how to do it, see Section 9.3.

To further conclude how this project contributes to the annotation sector, it provides a structure for approaching similar problems in the future. These design guidelines emphasize the importance of rotation micro-adjustments, feedback to annotators, improving existing tools over creating new ones, annotators' culture, and how micro-solutions can minimize micro-adjustments. Additionally, they can assist in projects with users located far away from where the study takes place, highlighting the significance of gathering information from multiple sources.

Bibliography

[1] D. Fagnant and K. Kockelman, "Preparing a Nation for Autonomous Vehicles: Opportunities, Barriers and Policy Recommendations," 2015, pp. 167–181. DOI: https://doi.org/10.1016/j.tra.2015.04.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0965856415000804.

[2] "Kognic - dataset management for sensor-fusion." (), [Online]. Available: https://www.kognic.com (visited on 01/29/2024).

[3] On-Road Automated Driving (ORAD) committee, "Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles," SAE International. DOI: 10.4271/J3016_202104. [Online]. Available: https://www.sae.org/content/j3016_202104 (visited on 01/23/2024).

[4] M. R. Bachute and J. M. Subhedar, "Autonomous driving architectures: Insights of machine learning and deep learning algorithms," *Machine Learning with Applications*, vol. 6, p. 100 164, 2021.

[5] A. Faisal, M. Kamruzzaman, T. Yigitcanlar, and G. Currie, "Understanding autonomous vehicles: A systematic literature review on capability, impact, planning and policy," *Journal of Transport and Land Use*, vol. 12, no. 1, pp. 45–72, 2019, ISSN: 19387849. [Online]. Available: https://www.jstor.org/stable/26911258 (visited on 01/23/2024).

[6] J. Manyika, M. Chui, J. Bughin, R. Dobbs, P. Bisson, and A. Marrs, *Disruptive technologies: Advances that will transform life, business, and the global economy*. McKinsey Global Institute San Francisco, CA, 2013, vol. 180.

[7] S. Singh, "Critical reasons for crashes investigated in the national motor vehicle crash causation survey," Tech. Rep., 2015.

[8] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 187–210, 2018.

[9] J. Wang, L. Zhang, Y. Huang, J. Zhao, and F. Bella, "Safety of autonomous vehicles," *Journal of advanced transportation*, vol. 2020, pp. 1–13, 2020.

[10] P. Jing, G. Xu, Y. Chen, Y. Shi, and F. Zhan, "The determinants behind the acceptance of autonomous vehicles: A systematic review," *Sustainability*, vol. 12, no. 5, p. 1719, 2020.

[11] C. J. Haboucha, R. Ishaq, and Y. Shiftan, "User preferences regarding autonomous vehicles," *Transportation Research Part C: Emerging Technologies*, vol. 78, pp. 37–49, 2017.

[12] K. Othman, “Public acceptance and perception of autonomous vehicles: A comprehensive review,” *AI and Ethics*, vol. 1, no. 3, pp. 355–387, 2021.

[13] J. Deichmann, E. Ebel, K. Heineke, R. Heuss, M. Kellner, and F. Steiner, “Autonomous drivings future: Convenient and connected,” McKinsey & Company, Jan. 6, 2023. [Online]. Available: `https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/autonomous-drivings-future-convenient-and-connected` (visited on 01/29/2024).

[14] S. D. Pendleton, H. Andersen, X. Du, *et al.*, “Perception, planning, control, and coordination for autonomous vehicles,” *Machines*, vol. 5, no. 1, p. 6, 2017.

[15] J. Pustejovsky and A. Stubbs, *Natural Language Annotation for Machine Learning: A guide to corpus-building for applications.* " O’Reilly Media, Inc.", 2012, pp. 1–31.

[16] “Deep learning vs. machine learning: A beginners guide,” Coursera. (Nov. 29, 2023), [Online]. Available: `https://www.coursera.org/articles/ai-vs-deep-learning-vs-machine-learning-beginners-guide` (visited on 02/05/2024).

[17] C. Janiesch, P. Zschech, and K. Heinrich, “Machine learning and deep learning,” *Electronic Markets*, vol. 31, no. 3, pp. 685–695, 2021.

[18] Y. Li and J. Ibanez-Guzman, “Lidar for autonomous driving: The principles, challenges, and trends for automotive lidar and perception systems,” *IEEE Signal Processing Magazine*, vol. 37, no. 4, pp. 50–61, 2020.

[19] C. Cui, Y. Ma, J. Lu, and Z. Wang, “Redformer: Radar enlightens the darkness of camera perception with transformers,” *IEEE Transactions on Intelligent Vehicles*, 2023.

[20] Y. Li, “Stereo vision and lidar based dynamic occupancy grid mapping: Application to scenes analysis for intelligent vehicles,” Ph.D. dissertation, Belfort-Montbéliard, 2013.

[21] J. Van Brummelen, M. OBrien, D. Gruyer, and H. Najjaran, “Autonomous vehicle perception: The technology of today and tomorrow,” *Transportation research part C: emerging technologies*, vol. 89, pp. 384–406, 2018.

[22] E. Brynjolfsson and T. Mitchell, “What can machine learning do? workforce implications,” *Science*, vol. 358, no. 6370, pp. 1530–1534, 2017.

[23] A. Petrovai, A. D. Costea, and S. Nedevschi, “Semi-automatic image annotation of street scenes,” in *2017 IEEE intelligent vehicles symposium (IV)*, IEEE, 2017, pp. 448–455.

[24] B. Wang, V. Wu, B. Wu, and K. Keutzer, “Latte: Accelerating lidar point cloud annotation via sensor fusion, one-click annotation, and tracking,” in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, IEEE, 2019, pp. 265–272.

[25] “What are wicked problems? updated 2024,” The Interaction Design Foundation. (Feb. 1, 2024), [Online]. Available: `https://www.interaction-design.org/literature/topics/wicked-problems` (visited on 01/29/2024).

[26] H. W. Rittel and M. M. Webber, “Dilemmas in a general theory of planning,” *Policy sciences*, vol. 4, no. 2, pp. 155–169, 1973.

[27] R. Buchanan, “Wicked problems in design thinking,” *Design issues*, vol. 8, no. 2, pp. 5–21, 1992.

[28] T. Brown *et al.*, "Design thinking," *Harvard business review*, vol. 86, no. 6, p. 84, 2008.
[29] J. C. Jones, *Design methods*. John Wiley & Sons, 1992, pp. 64–66.
[30] "Design sprint methodology," Design Sprints. (), [Online]. Available: `https://designsprintkit.withgoogle.com/methodology/overview` (visited on 01/29/2024).
[31] D. Norman and S. Draper, *User Centered System Design: New Perspectives on Human-computer Interaction* (New Perspectives on Human-Computer Interaction Series). Lawrence Erlbaum Associates, 1986, ISBN: 978-0-89859-781-3. [Online]. Available: `https://books.google.se/books?id=IR9FAAAAYAAJ`.
[32] T. Lowdermilk, *User-Centered Design: A Developer's Guide to Building User-Friendly Applications*. O'Reilly Media, Incorporated, 2013, p. 6, ISBN: 978-1-4493-5980-5. [Online]. Available: `https://books.google.se/books?id=XiX5bNJjWOkC`.
[33] C. Abras, D. Maloney-Krichmar, J. Preece, *et al.*, "User-centered design," *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications*, vol. 37, no. 4, pp. 445–456, 2004.
[34] J.-Y. Mao, K. Vredenburg, P. W. Smith, and T. Carey, "The state of user-centered design practice," *Communications of the ACM*, vol. 48, no. 3, pp. 105–109, 2005.
[35] T. Jokela, N. Iivari, J. Matero, and M. Karukka, "The standard of user-centered design and the standard definition of usability: Analyzing iso 13407 against iso 9241-11," in *Proceedings of the Latin American conference on Human-computer interaction*, 2003, pp. 53–60.
[36] A. Upadhyay and C. Kumbharana, "Perceptual study of behavioral implications of usage of ict for sustainable e-governance in rural india," *International Journal of Scientific and Research Publications*, vol. 2, no. 4, pp. 1–5, 2012.
[37] D. Diaper and N. Stanton, "The handbook of task analysis for human-computer interaction," pp. 6–21, 2003.
[38] A. Cooper, R. Reimann, D. Cronin, and C. Noessel, *About Face: The Essentials of Interaction Design*, 4th. Wiley Publishing, 2014, pp. 208–211, 243–247, ISBN: 1118766571.
[39] G. Lee, C. M. Eastman, T. Taunk, and C.-H. Ho, "Usability principles and best practices for the user interface design of complex 3d architectural design and engineering tools," *International journal of human-computer studies*, vol. 68, no. 1-2, pp. 90–104, 2010.
[40] R. Molich and J. Nielsen, "Improving a human-computer dialogue," *Communications of the ACM*, vol. 33, no. 3, pp. 338–348, 1990.
[41] A. Marczewski, *Gamification: a simple introduction*. Andrzej Marczewski, 2013, p. 4.
[42] E. D. Mekler, F. Brühlmann, A. N. Tuch, and K. Opwis, "Towards understanding the effects of individual gamification elements on intrinsic motivation and performance," *Computers in human behavior*, vol. 71, pp. 525–534, 2017.
[43] "APA dictionary of psychology." (Apr. 19, 2018), [Online]. Available: `https://dictionary.apa.org/motivation` (visited on 02/02/2024).

[44] R. M. Ryan and E. L. Deci, "Intrinsic and extrinsic motivations: Classic definitions and new directions," *Contemporary educational psychology*, vol. 25, no. 1, pp. 54–67, 2000.

[45] B. S. Frey and M. Osterloh, *Successful management by motivation: Balancing intrinsic and extrinsic incentives.* Springer Science & Business Media, 2001, pp. 7–10.

[46] T. M. Amabile, "Motivational synergy: Toward new conceptualizations of intrinsic and extrinsic motivation in the workplace," *Human resource management review*, vol. 3, no. 3, pp. 185–201, 1993.

[47] V. C. Ganta, "Motivation in the workplace to improve the employee performance," *International Journal of Engineering Technology, Management and Applied Sciences*, vol. 2, no. 6, pp. 221–230, 2014.

[48] R. O. Frost, P. Marten, C. Lahart, and R. Rosenblate, "The dimensions of perfectionism," *Cognitive therapy and research*, vol. 14, pp. 449–468, 1990.

[49] D. D. Burns, "The perfectionists script for self-defeat," *Psychology today*, vol. 14, no. 6, pp. 34–52, 1980.

[50] S. J. Ashford, R. Blatt, and D. VandeWalle, "Reflections on the looking glass: A review of research on feedback-seeking behavior in organizations," *Journal of management*, vol. 29, no. 6, pp. 773–799, 2003.

[51] A. Caraban, E. Karapanos, D. Gonçalves, and P. Campos, "23 ways to nudge: A review of technology-mediated nudging in human-computer interaction," in *Proceedings of the 2019 CHI conference on human factors in computing systems*, 2019, pp. 1–15.

[52] D. Kahneman, *Thinking, fast and slow.* Farrar, Straus and Giroux, 2011, ISBN: 978-0-374-27563-1. [Online]. Available: https://search.ebscohost.com/login.aspx?direct=true&db=cat09075a&AN=clpc.oai.edge.chalmers.folio.ebsco.com.fs00001000.75f98eb3.c04a.4adb.95af.7d51c511d7cc&site=eds-live&scope=site&authtype=guest&custid=s3911979&groupid=main&profile=eds.

[53] J. Randolph, "A guide to writing the dissertation literature review," *Practical assessment, research, and evaluation*, vol. 14, no. 1, p. 13, 2019.

[54] M. D. Gall, W. R. Borg, and J. P. Gall, *Educational research: An introduction.* Longman Publishing, 1996, pp. 161–162.

[55] K. Baxter, C. Courage, and K. Caine, *Understanding your users: a practical guide to user research methods.* Morgan Kaufmann, 2015, pp. 25–35.

[56] Å. Wikberg-Nilsson, Å. Ericson, and P. Törlind, *Design: process och metod.* Studentlitteratur AB, 2015, pp. 45, 85.

[57] "How to conduct user observations," The Interaction Design Foundation. (Mar. 24, 2021), [Online]. Available: https://www.interaction-design.org/literature/article/how-to-conduct-user-observations (visited on 02/01/2024).

[58] J. C. Tang, S. B. Liu, M. Muller, J. Lin, and C. Drews, "Unobtrusive but invasive: Using screen recording to collect field data on computer-mediated interaction," in *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*, 2006, pp. 479–482.

[59] L. Bao, J. Li, Z. Xing, X. Wang, X. Xia, and B. Zhou, "Extracting and analyzing time-series hci data from screen-captured task videos," *Empirical Software Engineering*, vol. 22, pp. 134–174, 2017.
[60] "What are design requirements?" The Interaction Design Foundation. (), [Online]. Available: `https://www.interaction-design.org/literature/topics/design-requirements` (visited on 01/31/2024).
[61] B. E. John and D. E. Kieras, "Using goms for user interface design and evaluation: Which technique?" *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 3, no. 4, pp. 287–319, 1996.
[62] D. Cunha, R. P. Duarte, and C. A. Cunha, "Klm-goms detection of interaction patterns through the execution of unplanned tasks," in *International conference on computational science and its applications*, Springer, 2021, pp. 203–219.
[63] D. Kieras *et al.*, "Using the keystroke-level model to estimate execution times," *University of Michigan*, vol. 555, 2001.
[64] "What are semi-structured interviews?" The Interaction Design Foundation. (), [Online]. Available: `https://www.interaction-design.org/literature/topics/semi-structured-interviews` (visited on 02/01/2024).
[65] N. Pacaol, "Teacher's workload intensification: A qualitative case study of its implications on teaching quality.," *International Online Journal of Education and Teaching*, vol. 8, no. 1, pp. 43–60, 2021.
[66] H. Kallio, A.-M. Pietilä, M. Johnson, and M. Kangasniemi, "Systematic methodological review: Developing a framework for a qualitative semi-structured interview guide," *Journal of advanced nursing*, vol. 72, no. 12, pp. 2954–2965, 2016.
[67] K. Kaplan. "5 ux workshops and when to use them: A cheat sheet," Nielsen Norman Group. (Aug. 14, 2023), [Online]. Available: `https://www.nngroup.com/articles/5-ux-workshops/` (visited on 02/01/2024).
[68] R. Ørngreen and K. T. Levinsen, "Workshops as a research methodology," *Electronic Journal of E-learning*, vol. 15, no. 1, pp. 70–81, 2017.
[69] M. Tomitsch, C. Wrigley, M. Borthwick, *et al.*, *Design. think. make. break. repeat. A handbook of methods.* BIS publishers, 2018.
[70] "Affinity diagrams," The Interaction Design Foundation. (), [Online]. Available: `https://www.interaction-design.org/literature/topics/affinity-diagrams` (visited on 02/01/2024).
[71] A. Lucero, "Using affinity diagrams to evaluate interactive prototypes," in *Human-Computer Interaction–INTERACT 2015: 15th IFIP TC 13 International Conference, Bamberg, Germany, September 14-18, 2015, Proceedings, Part II 15*, Springer, 2015, pp. 231–248.
[72] "Design requirements," The Interaction Design Foundation. (), [Online]. Available: `https://www.interaction-design.org/literature/topics/design-requirements` (visited on 01/31/2024).
[73] "Making your UX life easier with the MoSCoW," The Interaction Design Foundation. (Nov. 18, 2015), [Online]. Available: `https://www.interaction-design.org/literature/article/making-your-ux-life-easier-with-the-moscow` (visited on 01/31/2024).

[74] A. Harley. "Ideation for everyday design challenges," Nielsen Norman Group. (Jan. 15, 2017), [Online]. Available: https://www.nngroup.com/articles/ux-ideation/ (visited on 01/31/2024).
[75] "Ideation for design - preparing for the design race," The Interaction Design Foundation. (Aug. 2, 2020), [Online]. Available: https://www.interaction-design.org/literature/article/ideation-for-design-preparing-for-the-design-race (visited on 01/31/2024).
[76] "Brainstorming," The Interaction Design Foundation. (), [Online]. Available: https://www.interaction-design.org/literature/topics/brainstorming (visited on 01/31/2024).
[77] A. F. Osborn, "Applied imagination.," 1953.
[78] J. Rojas. "Learn how to use sketching as an ideation method," The Interaction Design Foundation. (Sep. 14, 2023), [Online]. Available: https://www.interaction-design.org/literature/article/etch-a-sketch-how-to-use-sketching-in-user-experience-design (visited on 01/31/2024).
[79] K. Pernice. "Design charrettes & team sketching: $\frac{1}{2}$ Inspiration, $\frac{1}{2}$ buy-in," Nielsen Norman Group. (Dec. 22, 2013), [Online]. Available: https://www.nngroup.com/articles/design-charrettes/ (visited on 01/31/2024).
[80] K. Pernice. "UX prototypes: Low fidelity vs. high fidelity," Nielsen Norman Group. (Dec. 22, 2016), [Online]. Available: https://www.nngroup.com/articles/ux-prototype-hi-lo-fidelity/ (visited on 01/31/2024).
[81] "Wireframing," The Interaction Design Foundation. (), [Online]. Available: https://www.interaction-design.org/literature/topics/wireframing (visited on 01/31/2024).
[82] "High-fidelity prototyping: What is it and how can it help?" Figma. (), [Online]. Available: https://www.figma.com/resource-library/high-fidelity-prototyping/ (visited on 01/31/2024).
[83] "Figma: The collaborative interface design tool," Figma. (), [Online]. Available: https://www.figma.com/ (visited on 01/31/2024).
[84] M. Soegaard. "User experience (UX) surveys: The ultimate guide," The Interaction Design Foundation. (Oct. 6, 2023), [Online]. Available: https://www.interaction-design.org/literature/article/ux-surveys (visited on 01/31/2024).
[85] "A/b testing," The Interaction Design Foundation. (2024), [Online]. Available: https://www.interaction-design.org/literature/topics/a-b-testing (visited on 04/05/2024).
[86] C. Lewis and C. Wharton, "Chapter 30 - cognitive walkthroughs," in *Handbook of Human-Computer Interaction (Second Edition)*, M. G. Helander, T. K. Landauer, and P. V. Prabhu, Eds., Second Edition, Amsterdam: North-Holland, 1997, pp. 717–732, ISBN: 978-0-444-81862-1. DOI: https://doi.org/10.1016/B978-044481862-1.50096-0. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780444818621500960.
[87] *What is Cognitive Walkthrough?* May 25, 2026. [Online]. Available: https://www.interaction-design.org/literature/topics/cognitive-walkthrough (visited on 04/03/2024).

[88] M. H. Blackmon, P. G. Polson, M. Kitajima, and C. Lewis, "Cognitive walkthrough for the web," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '02, Minneapolis, Minnesota, USA: Association for Computing Machinery, 2002, pp. 463–470, ISBN: 1581134533. DOI: 10.1145/503376.503459. [Online]. Available: https://doi.org/10.1145/503376.503459.

[89] M. Brown. "Writing good survey questions: 10 best practices," Nielsen Norman Group. (Aug. 3, 2023), [Online]. Available: https://www.nngroup.com/articles/survey-best-practices/ (visited on 01/31/2024).

[90] "Kognic: Fair working conditions & global dependencies in machine learning." (Jun. 15, 2022), [Online]. Available: https://www.kognic.com/articles/fair-working-conditions (visited on 02/02/2024).

[91] "Code of conduct." (), [Online]. Available: https://www.kognic.com/code-of-conduct (visited on 02/02/2024).

[92] "General data protection regulation GDPR," General Data Protection Regulation (GDPR). (), [Online]. Available: https://gdpr-info.eu/ (visited on 02/02/2024).

[93] W. L. i. R.-B. U. Experience. "Nielsen norman group: UX training, consulting, & research," Nielsen Norman Group. (), [Online]. Available: https://www.nngroup.com/ (visited on 03/06/2024).

[94] "UX design courses & global UX community," The Interaction Design Foundation. (), [Online]. Available: https://www.interaction-design.org (visited on 03/06/2024).

[95] Miro, *Miro*, version 2.o, Mar. 14, 2024. [Online]. Available: https://miro.com/.

[96] P. Woniak, M. Fjeld, and S. Zhao, "Limiting trial and error: Introducing a systematic approach to designing clutching," in *Proceedings of the Second International Symposium of Chinese CHI*, Toronto Ontario Canada: ACM, Apr. 26, 2014, pp. 35–39, ISBN: 978-1-4503-2876-0. DOI: 10.1145/2592235.2592241. [Online]. Available: https://dl.acm.org/doi/10.1145/2592235.2592241 (visited on 06/05/2024).

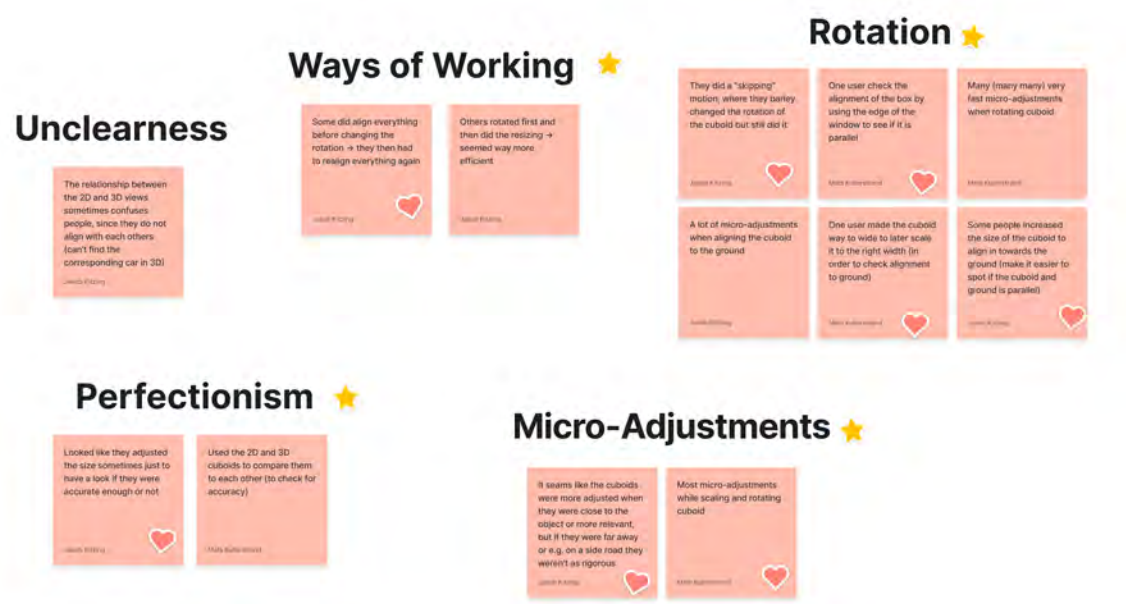
[97] A. M. Feit, S. Williams, A. Toledo, *et al.*, "Toward everyday gaze input: Accuracy and precision of eye tracking and implications for design," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, Denver Colorado USA: ACM, May 2, 2017, pp. 1118–1130, ISBN: 978-1-4503-4655-9. DOI: 10.1145/3025453.3025599. [Online]. Available: https://dl.acm.org/doi/10.1145/3025453.3025599 (visited on 06/05/2024).

A

Appendix

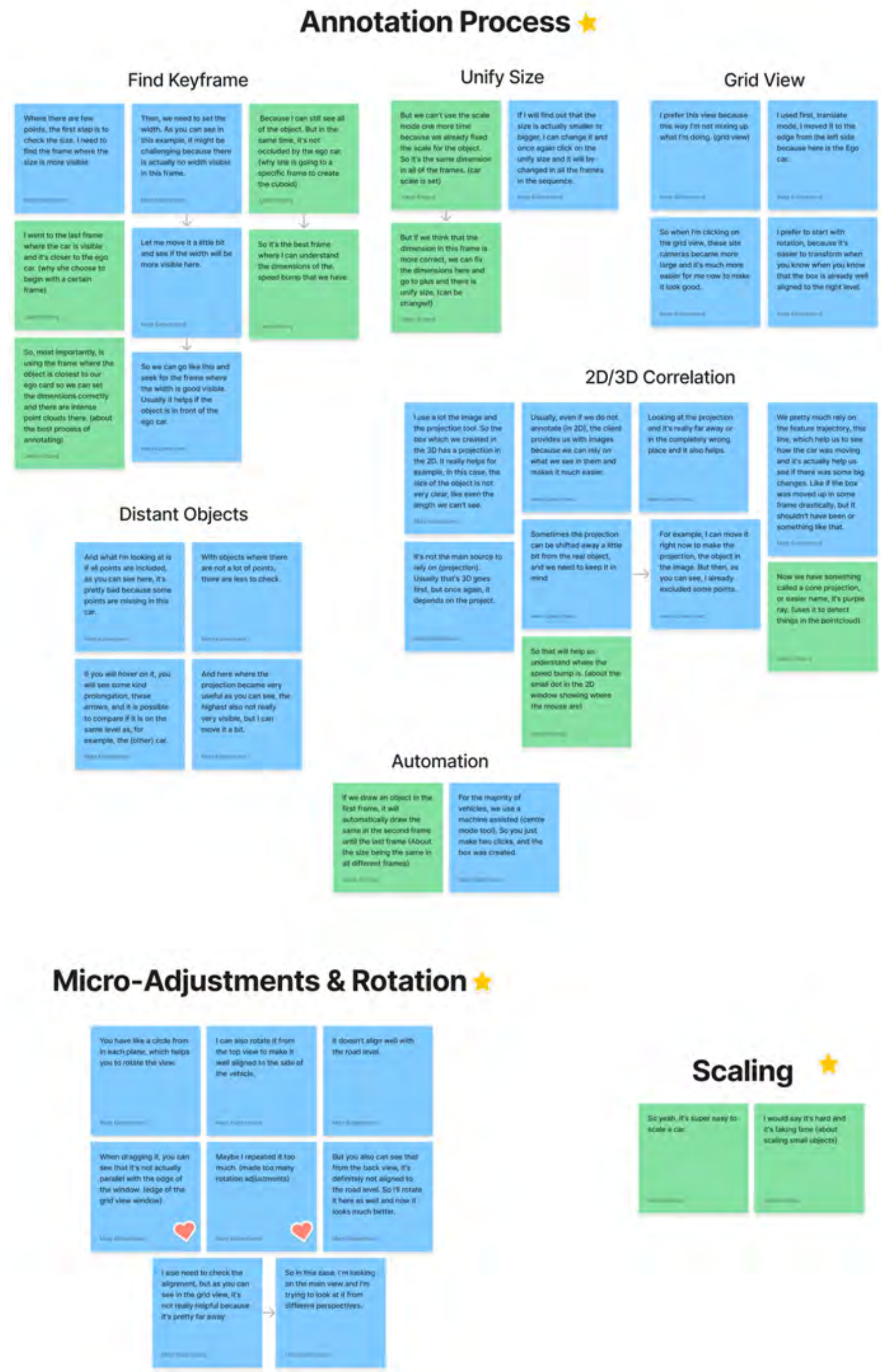
A.1 Affinity Diagramming - Observations

This appendix shows the affinity diagramming from the observations. The most important categories extracted were: ways of working, rotation, perfectionism, and micro-adjustments.



A.2 Affinity Diagramming - Semi-structured Interviews

This appendix shows the affinity diagramming from the semi-structured interviews. The most important categories extracted were: ways of working, perfectionism, micro-adjustment, expert users, and task guidelines.

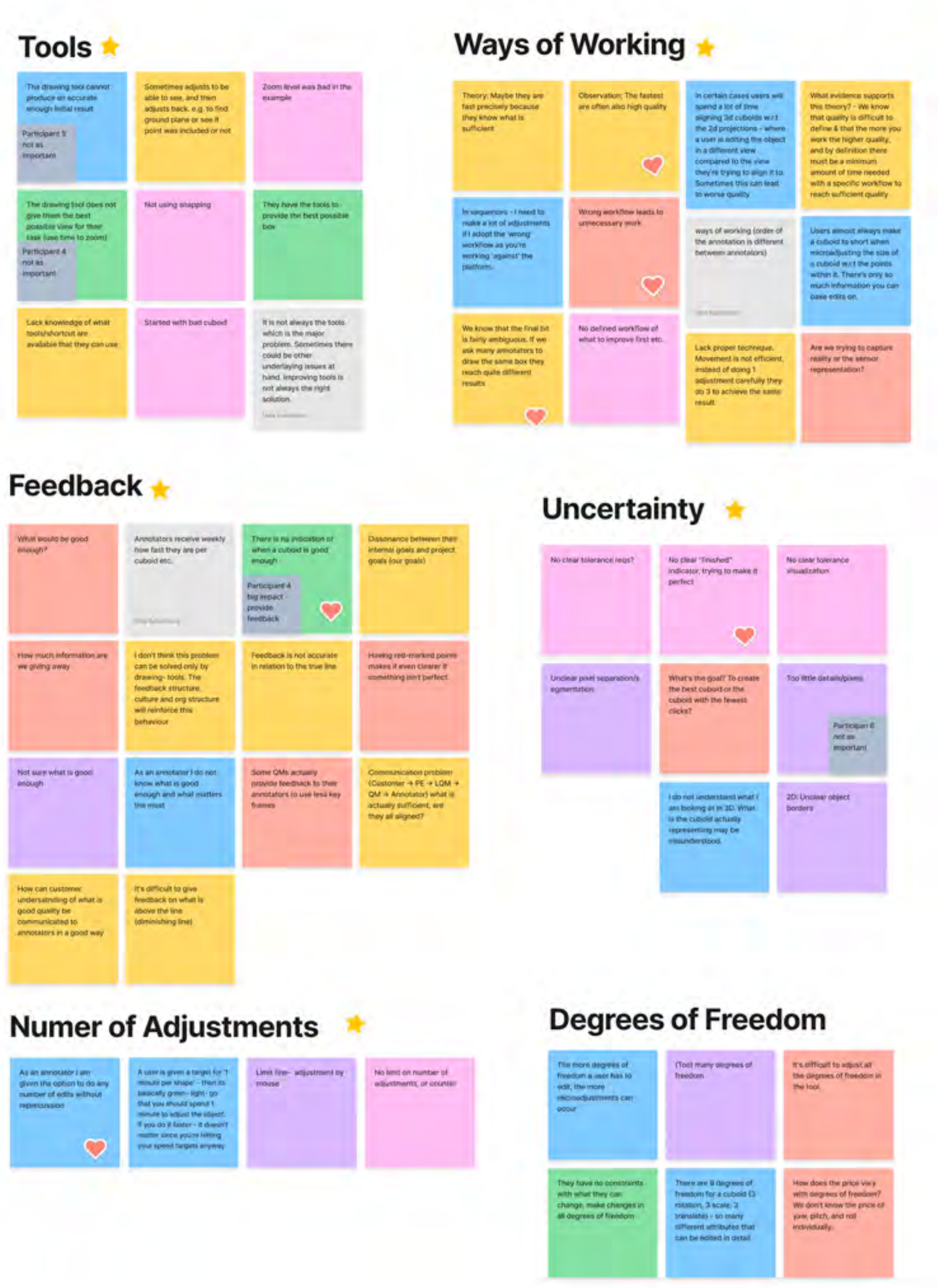


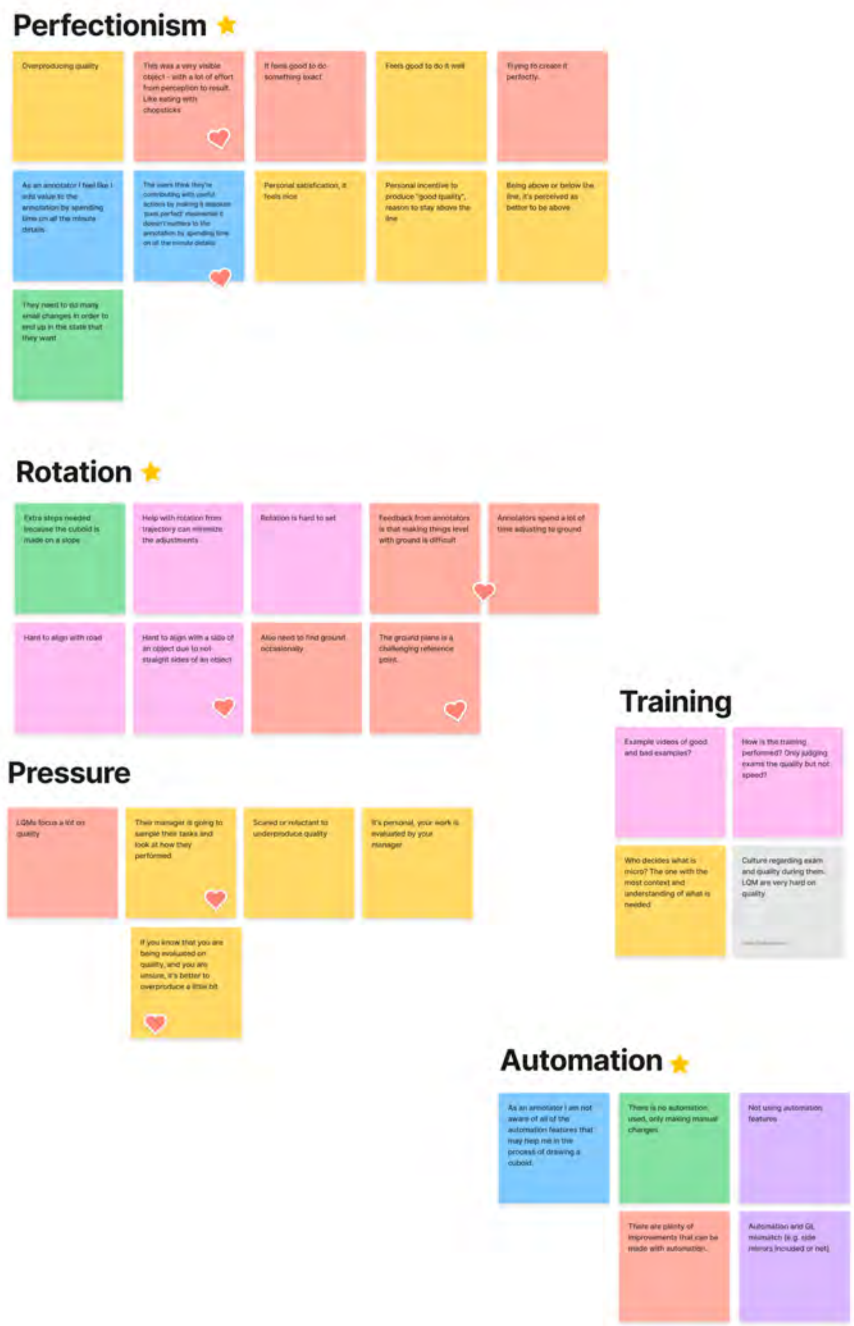

Ways of Working
I try different way of annotation, and I check which is faster, or I reach out to colleagues that I think have more experience than me in this, area, and ask them for help. (how she knows which process is the best)
Perfectionism
Here is the last boundary of the point, so I have to fix it one more time, like this. (Describing what she is doing)
As many times as I see it's convenient and it's sorry, it's correct. (describes how she does micro-adjustments as many times as needed to make it correct)
The bounding box should be as tight as possible, and it should only include the points of the object.
I needed to make the box as tight as possible and include only the point clouds for that object.
It's just as close as possible if there are enough points.
So that's the boundaries of the speed bump that I have. The box should be as tight as possible.
So as you can see, I'm looking, fixing it, looking one more time and fixing. (describes that she is doing micro-adjustments)
If my goal is to create the perfect annotation, I can do even some really small adjustments, like move it for a few millimetres
But at the same time, this is not acceptable, as you can see from here, but this is acceptable, or even this one is acceptable (puts the line directly besides the dots, and then changes so that the line is directly on the dots).
New Tools
Sometimes there are videos but either way we can check it in the user manual user documentation (new tools)
Usually they are just writing in Slack, it is how it was before. But now we also have notifications in app on the dashboard. So if I will come to the dashboard. (new features)
Hard Annotating
This white line, is the trajectory from the ego car, but it's not always mean that it should be on the same level as the car because the road has ups and downs and it's not always the source to rely on.
The front and the right sides are not close to the car, so that's why the image of the sides is not clear.
Firstly, if the object was in a distance (about what is hard with annotating)
It can be challenging when there are not a lot of points.
Accuracy of this correct stuff, which I'm talking about also depends on the projects because, it's easier to take an example from the 2D because there are pixels in the images.
This car is really far away, there's no way to tell me, to find out, the level in the grid view.
Sometimes it's hard for us to understand where this object location is. So we have to relay on another functions in the project. For example, we can check the distance from a static object. (what's hard with annotating)
There are like only a few points and I think they belong to this car. So to like create the perfect box for this car will be quite, quite challenging.
Because we do not have enough pointy clouds to annotate this object. (the hardest part about annotating)
Easy Annotating
With boxes, it's the most easy thing and usually if you have a road or a highway. There will be basically vehicles and they are pretty close to the ego car
The easiest thing to annotate is the objects which are very close to the ego car
For me, it's pretty easy, even if the object is far away or heavily polluted or something like this.
And many of annotators, especially if they lack experience, will not see it.
You need to waste your time on it, but at least it's not hard for me to understand that it's somewhere there, and to mark it, and yeah, it will take time, and it's not very funny, but yeah, at least I know that it is there.
Fun Annotating
I prefer to annotate then pick on someone's work, but just to make a good work by myself.
It's pretty cool for me, to know where the farthest objects are, or to understand where is the bush and where is the car, because in 3D it can be pretty much challenging.
Collaboration
From what we can see in their tasks, they are helping each other, especially maybe some more experienced users or PMs.
We had a lot of discussions. So, I guess it's the same with all the teams. Like if there's a person who knows better, this person helps. So, anyone who need it.
We were annotating also, so we were discussing different things, and we were like conflicting with each other, like I am right, it should be done like this. No, I am right, and it should be done differently, or something like that.
Training
Once some pass exam and we provided the feedback, there are other users are doing the same stuff and not always it's the correct stuff.
They are repeating the same mistake and they are like copying each other work, which can be the issue. But in the same time, it's good if they help each other.
Let's say, bounding boxes took me three months to say that I fully understand how to annotate objects. (how hard it is to learn how to annotate)
Based on my knowledge here, I can provide some instructions or videos or schemes to help annotators with it.
But other features like 3D line and lanes, it takes less time, maybe one month. (about how long it takes to master lanes annotation

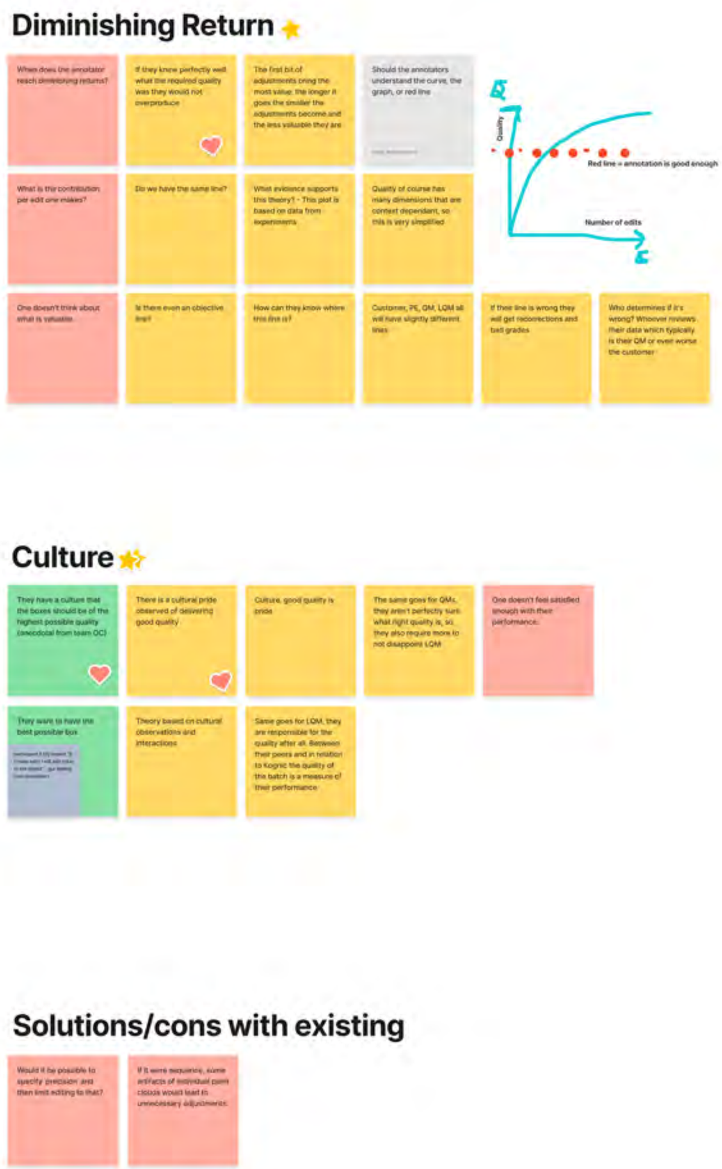

Task Guidelines
They, decide how many pixels we have to tolerate, but what is acceptable is not leaving any pixels unnecessarily within the box. (about the guidelines)
So, I rely on the guideline and the dimensions of the cars provided in that guideline. (If it's not possible to know how long a certain car is)
When we look at the 2D image and the 3D box should cover all the 2D polygons that we drew.
If the object is close, it's obvious and it's a mistake
All projects have strict requirements about including road points or excluding it.
Once again, depends on the requirements for each project. (each project has different requirements)
For example, if you're making a mistake, like 10 centimetres or 20 centimetres in size or in position.
The further away the object is, the bigger mistake you can do. It will be acceptable because from this distance, it's impossible to say if it's actually a correct level.
Some projects has a requirement that the box should be tight as possible to the object, but, plus, minus two pixels are okay(2D)
We have requirements.
For example, we do not include side mirrors, but all other points should be included.
In 3D, I think we have some similar, but there are no clear requirements because there are no pixels or something like that.
I really like when the client do this, we can have some default sizes.
It means that some of the parameters is ambiguous and it's hard to set it. We can just use the default size, which they provided in their guideline or separate document.
They allow us two centimetres, plus minus two centimetres. So even if I'm using like 178, it still can be found as a default size.
Expert Annotators
So the annotators are just annotators.
For more than four years. (work experience)
There's another type of task which calls quality assurance tasks. In this case, we check and fix only the targeted frames.
So the work of LQM is to track the quality and speed and to detect mistakes and help the team to improve them.
If there are missed objects, I need to annotate them. (LQM)
If I move the box, it will change the position. So it means that the position of the annotator was slightly incorrect. This mistake will be recorded in the IQR, a quality report.
Our work is to check if everything was done correctly. (LQM)
The most general tasks we are working with are correction tasks (LQM)
Sometimes they are mixing up the class or the default size property is set as yes.
We can also do some regular kind of feedback, like we are checking tasks and noticing that some mistakes are repeating. And in this case, we're just creating a Google Doc and we are putting the mistake in there. Describing it and showing how it should be done correctly. So it's not enough just to tell it was wrong.
Quality assurance tasks, helps to save the time so we do not need to check all 78 frames, but only a few.
So I'm like pretty much doing the job, which annotators should have done to make it look good, but like, yeah, I'm making it look good
We just leave in the comments what was done wrong and how it should be fixed. And then we reject in the task and it comes back to the user who did it. This way the user can learn.
To be honest, I'm not sure about it. I think it should be a little bit higher; just a tiny bit, at the same time, it is pretty acceptable.
In case annotators need more detailed explanation, we can also have a meeting with them or record the video to show the process fully or something like this.
Or if it's a review, I will just leave a comment. If it's the correction, I will correct and make it look how it should be. (different types of tasks)
If I saw some mismatching anywhere, I will fix it.
First is to be sure that the size is correct. Then to make sure that the alignment to the ground level is correct. Then look all the properties if they are correct.
Depending on which type of task it is, like in the quality assurance task, we do not need to do unnecessary changes, like really small adjustments. So I will leave it like that.

A.3 Affinity Diagramming - Workshop

This appendix shows the affinity diagramming from the workshop. The most important categories extracted were: culture, ways of working, uncertainty and feedback, diminishing return, perfectionism, tools, number of adjustments, rotation, and automation.

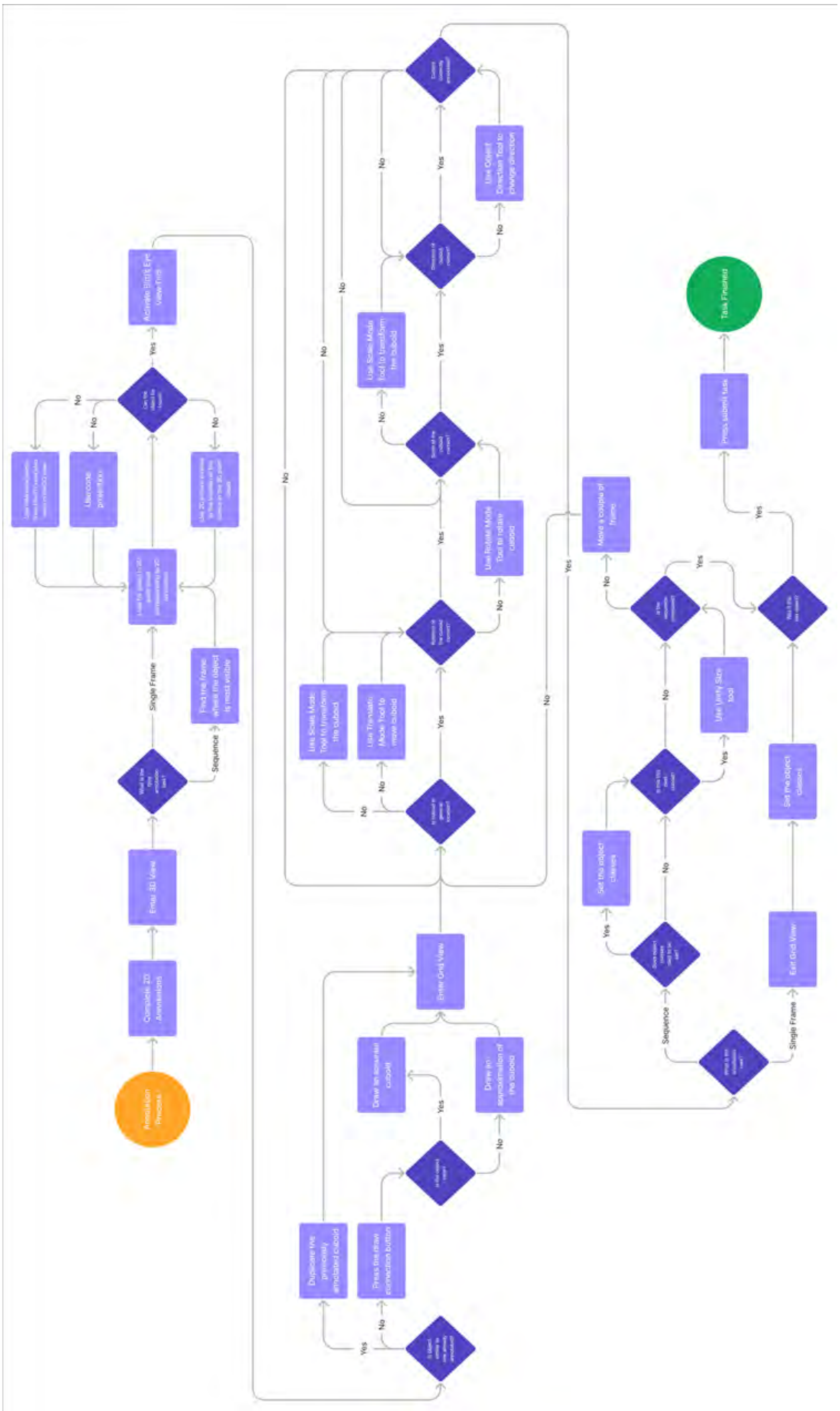



Perfectionism
Overproducing quality
This was a very visible object - with a lot of effort from perception to result. Like eating with chopsticks
It feels good to do something exact
Feels good to do it well
Trying to create it perfectly.
As an annotator I feel like I add value to the annotation by spending time on all the minute details
Personal satisfaction, it feels nice
Personal incentive to produce "good quality", reason to stay above the line
Being above or below the line, it's perceived as better to be above
They need to do many small changes in order to end up in the state that they want
Rotation
Extra steps needed because the cuboid is made on a slope
Help with rotation from trajectory can minimize the adjustments
Rotation is hard to set
Feedback from annotators is that making things level with ground is difficult
Annotators spend a lot of time adjusting to ground
Hard to align with road
Hard to align with a side of an object due to not straight sides of an object
Also need to find ground occasionally
The ground plane is a challenging reference point.
Pressure
LQMs focus a lot on quality
Their manager is going to sample their tasks and look at how they performed
Scared or reluctant to underproduce quality
It's personal, your work is evaluated by your manager
If you know that you are being evaluated on quality, and you are unsure, it's better to overproduce a little bit
Training
Example videos of good and bad examples?
How is the training performed? Only judging exams the quality but not speed?
Who decides what is micro? The one with the most context and understanding of what is needed
Culture regarding exam and quality during them. LQM are very hard on quality
Automation
As an annotator I am not aware of all of the automation features that may help me in the process of drawing a cuboid.
There is no automation used, only making manual changes
Not using automation features
There are plenty of improvements that can be made with automation.
Automation and GL mismatch (e.g. side mirrors included or not)


Diminishing Return
When does the annotator reach diminishing returns?
If they knew perfectly well what the required quality was they would not overproduce
The first bit of adjustments bring the most value: the longer it goes the smaller the adjustments become and the less valuable they are
Should the annotators understand the curve, the graph, or red line
Quality
Red line = annotation is good enough
Number of edits
What is the contribution per edit one makes?
Do we have the same line?
What evidence supports this theory? - This plot is based on data from experiments
Quality of course has many dimensions that are context dependant, so this is very simplified
One doesn't think about what is valuable
Is there even an objective line?
How can they know where this line is?
Customer, PE, QM, LQM all will have slightly different lines
If their line is wrong they will get redcorrections and bad grades
Who determines if it's wrong? Whoever reviews their data which typically is their QM or even worse the customer
Culture
They have a culture that the boxes should be of the highest possible quality (anecdotal from team OC)
There is a cultural pride observed of delivering good quality
Culture, good quality is pride
The same goes for QMs, they aren't perfectly sure what right quality is, so they also require more to not disappoint LQM
One doesn't feel satisfied enough with their performance.
They want to have the best possible box
Theory based on cultural observations and interactions
Same goes for LQM, they are responsible for the quality after all. Between their peers and in relation to Kognic the quality of the batch is a measure of their performance
Solutions/cons with existing
Would it be possible to specify 'precision' and then limit editing to that?
If it were sequence, some artifacts of individual point clouds would lead to unnecessary adjustments.

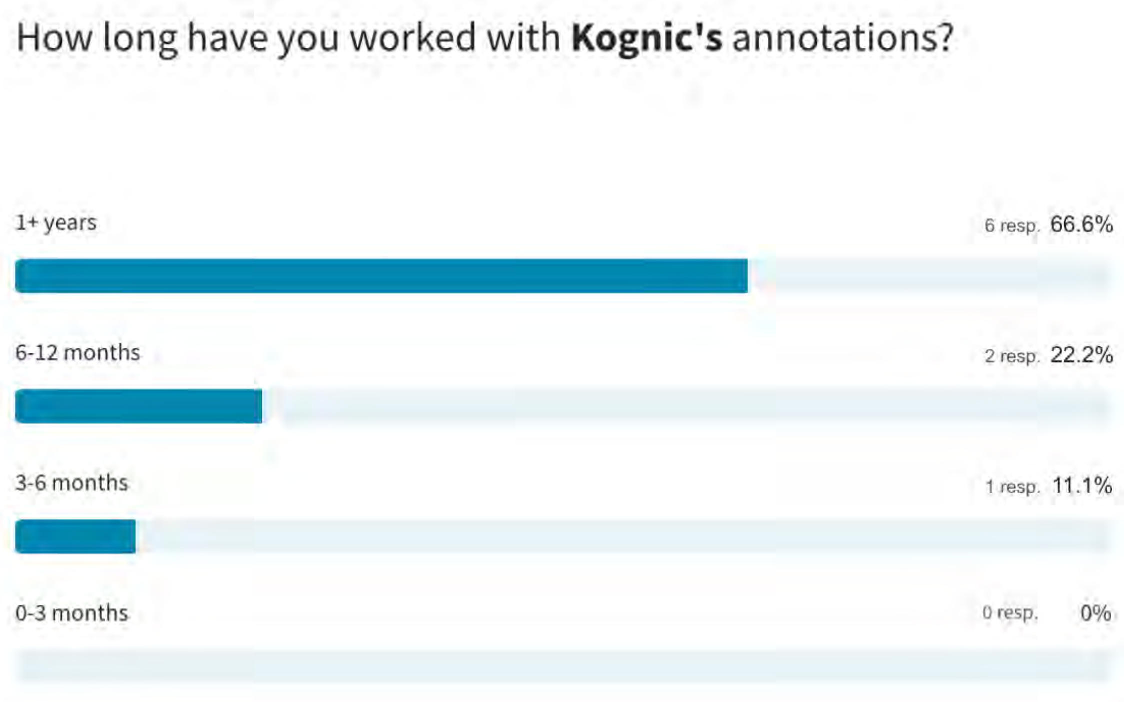
A.4 Flowchart

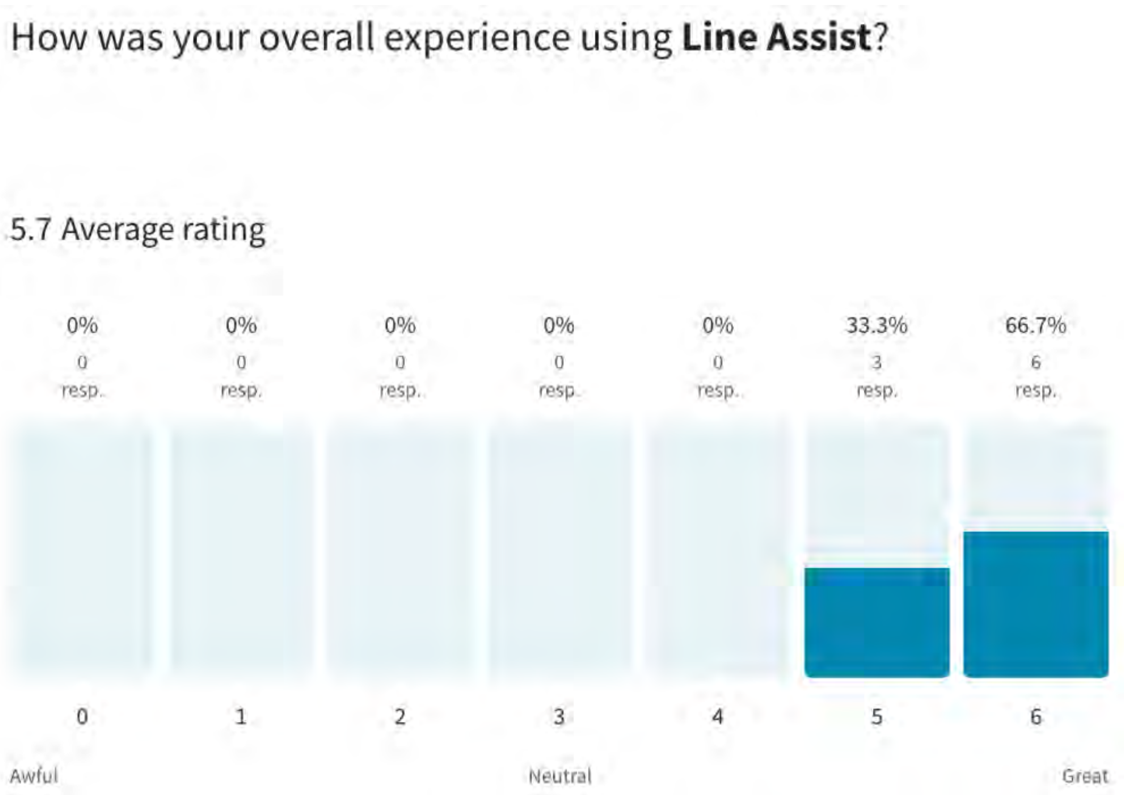
This appendix shows the entire flowchart of the process of annotation.



A.5 User Experience Survey

This appendix shows the results of the user experience survey used for evaluating Line Assist.



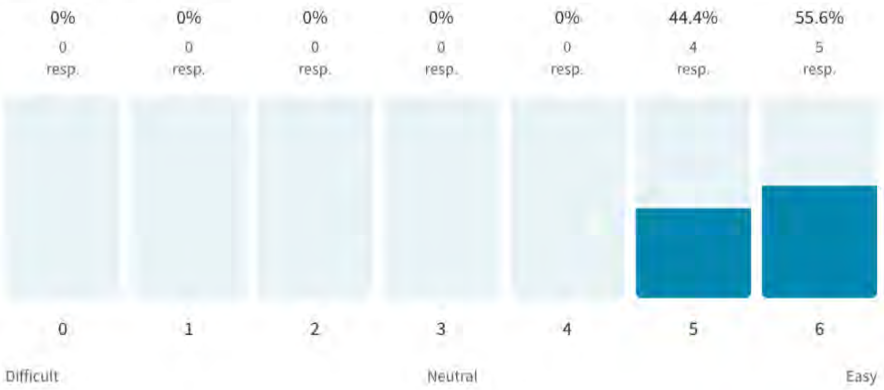


My experience with **Line Assist**

	Awful	Bad	Neutral	Good	Great
Editing comfortably	0%	0%	0%	100%	0%
Editing efficiently	0%	0%	11.1%	77.8%	11.1%
Editing accurately	0%	0%	11.1%	77.8%	11.1%
Learning how to use the tool	0%	0%	0%	77.8%	22.2%

How easy was it to align the rotation of the cuboid to the ground using **Line Assist**?

5.6 Average rating



A.6 Cognitive Walkthrough

The appendix shows the entire cognitive walkthrough conducted for the Sequence Process Wizard.

	Will the user try to achieve the right outcome?	**Will the user notice that the correct action is available to them?**	**Will the user associate the correct action with the outcome they expect to achieve?**	**If the correct action is performed, will the user see that progress is being made towards their intended outcome?**
Center Mode Tool	Yes - the user knows that the goal is to annotate objects	Yes - The user will see the center mode tool which includes a tooltip	Yes	Yes
Click on Tab	Yes	No - The user would not know how to move to the next step in the wizard	Yes - Tab means to jump, which correlates to the process wizard	Yes
Translate	No - The users might want to use their own process for adjusting the cuboid.	Yes	Yes	Yes
Rotate	No - The users might want to use their own process for adjusting the cuboid.	Yes	Yes	Yes

Scale	No - The users might want to use their own process for adjusting the cuboid.	Yes	Yes	Yes
Click on Tab	Yes - They want to move to the next step in the process	No - The user would not know that the action is available if they have not been taught that it is	Yes - Tab means to jump, which correlates to the process wizard	Yes
Trim Visibility	Yes	Yes	No - The user might only think they should mark out the beginning point since that is the icon used for this stage. This could lead them to move on to the next step without finishing trimming the visibility.	Yes
Translate	No - The users might want to use their own process for adjusting the cuboid.	Yes	Yes	Yes

Rotate	No - The users might want to use their own process for adjusting the cuboid.	Yes	Yes	Yes
Next Frame	Yes - when they have finalized a cuboid they want to go to the next frame	Yes	No - the user might think they only will jump one frame but in reality, the jumped five	No - The wizard in just goes back to the beginning
Click on Tab	No - the user might not understand they have finished all frames since they would want to move on to the next one according to the wizard	No - The user would not know how to move to the next step in the wizard	Yes - Tab means to jump, which correlates to the process wizard	Yes
Check Frames	Yes	Yes	Yes	Yes