Characterizing Annotations for Visual Analytics on Clinical Data

Dissertation

A thesis submitted for the degree of

Doktor-Ingenieur (Dr.-Ing.)

Faculty of Computer Science and Electrical Engineering
University of Rostock

Christoph Schmidt
born on July 9th 1978 in Rostock
resident in Schwerin

Rostock, November 12, 2024

https://doi.org/10.18453/rosdok_id00005035

Principal Advisor

▷ Prof. Dipl.-Ing. Dr.techn. Stefan Bruckner
University of Rostock, Germany

Reviewer

▷ Univ.-Prof. Dr.-Ing. Lars Linsen
University of Münster, Germany

▷ Prof. Dr. Paul Rosenthal
University of Rostock, Germany

Date of Presentation

▷ 7th of November 2024

Keywords
Annotation, Data Preprocessing, Data Cleansing, Data Exploration, Visual Analytics, Heterogeneous Clinical Data

Classification (ACM CCS 2012)

Human-centered computing (HCC)
HCC → Visualization → Visualization application domains → Visual analytics
HCC → Human computer interaction (HCI) → HCI theory, concepts, and models
HCC → Visualization → Visualization theory, concepts, and paradigms

Date of Submission: 24.04.2024

“Mut heißt nicht keine Angst zu haben. Mut heißt nur, dass man trotzdem springt.”

Sarah Lesch

Dedicated to my beloved family.

Acknowledgements

I started my scientific career at the University of Rostock with a presentation named "from a crime prosecutor to a scientist - always with the user in the loop." This metamorphosis has taken its time and culminates with this thesis. In no way I understood the meaning and consequences of this process at the start. The tremendous effort involved could only be done with the support of many. It is my sincere wish to point this out.

First of all, my supervisor Heidrun Schumann has the largest share on my transformation to a scientist. It was your work to patiently teach me the means of scientific work. There is no overstating. The plethora of constructive discussions, both on my scientific topic and politics were, are, and will be of great value for me. Nonetheless, this work could not have been finished, without my supervisor Stefan Bruckner's generous support and willingness to invest a lot of work. Your constructive advice made the finalization of this work possible.

Being able to finish a thesis, first of all needs a start. This start would not have been possible without Paul Rosenthal. You gave me the hint about the doctoral position at the university and thus the necessary push to apply. This made my long-held dream of scientific work come true. Thank you for your valuable advice throughout the whole doctoral process. My gratitude also goes to Martin Röhlig who started as a colleague and became a friend and like-minded pal in many ways. You made my start easy and made me feel better after another devastating supervisor consultation. You, and Philip Berger, Christian Eichner, Martin Luboschik, Christian Tominski, Steve Dübel and Christian Brode-Richter made the work and live at the university special, enlightening, and enjoyable. In this respect, a special reference goes to our great breakfast meetings.

I would like to thank my partners in the TOPOs project, especially Bastian Grundel and Andreas Stahl. Your patient support in providing the data, requirements and feedback made this work possible. My very special appreciation goes to the University of Grenada, who gave me shelter to focus on this work in times of rough sea and sunny beaches.

Finally, I would like to thank the ones I owe the most. There are no words to properly express the incredible patience, understanding, and support, my family gave. Manuela, Theodor, and Alfred - thank you. I love you so much.

List of Publications

M. Röhlig, P. Rosenthal, C. Schmidt, H. Schumann and O.Stachs. "Visual Analysis of Optical Coherence Tomography Data in Ophthalmology". In: *Proceedings of the EuroVis Workshop on Visual Analytics.* 2017, pp. 37-41.
DOI: 10.2312/eurova.20171117.

C. Schmidt, P. Rosenthal and H. Schumann. "Annotations as a Support for Knowledge Generation - Supporting Visual Analytics in the Field of Ophthalmology". In: *Proceedings of VISIGRAPP.* 2018, pp. 264-272.
DOI: 10.5220/0006615902640272.

M. Röhlig, C. Schmidt, R. K. Prakasam, H. Schumann, and O. Stachs. "Visual Analysis of Retinal Changes with Optical Coherence Tomography". In: *The Visual Computer* 34.9 (2018), pp. 1209-1224.
DOI: 10.1007/s00371-018-1486-x.

C. Schmidt, M. Röhlig, B. Grundel, P. Daumke, M. Ritter, A. Stahl, P. Rosenthal, and H. Schumann. "Combining Visual Cleansing and Exploration for Clinical Data". In: *Proceedings of the IEEE Workshop on Visual Analytics in Healthcare.* 2019, pp. 25-32.
DOI: 10.1109/VAHC47919.2019.8945034.

M. Röhlig, J. Stüwe, C. Schmidt, R. Prakasam, O. Stachs, and H. Schumann. "Grid-Based Exploration of OCT Thickness Data of Intraretinal Layers". In: *Proceedings of VISIGRAPP.* 2019, pp. 129-140.
DOI: 10.5220/0007580001290140.

M. Röhlig, C. Schmidt, R. K. Prakasam, O. Stachs, and H. Schumann. "Towards Accurate Visualization and Measurement of Localized Changes in Intraretinal Layer Thickness". In: *Proceedings of the IEEE Workshop on Visual Analytics in Healthcare.* 2019, pp. 58-59.
DOI: 10.1109/vahc47919.2019.8945028.

M. Röhlig, R. K. Prakasam, J. Stüwe, C. Schmidt, O. Stachs, and H. Schumann. "Enhanced Grid-Based Visual Analysis of Retinal Layer Thickness with Optical Coherence Tomography". In: MDPI Information 10.9 (2019), pp. 266:1-266:23.
DOI: 10.3390/info10090266.

C. Schmidt, P. Rosenthal, and H. Schumann. "Varying Annotations in the Steps of the Visual Analysis". 2020.
DOI: 10.48550/arXiv.2008.08806.

B. Grundel, M.-A. Bernardeau, H. Langner, C. Schmidt, D. Böhringer, M. Ritter, P. Rosenthal, A. Grandjean, S. Schulz, P. Daumke, and A. Stahl. "Merkmalsextraktion aus klinischen Routinedaten mittels Text-Mining". In: *Der Ophthalmologe* 118.3 (2020), pp. 264-272.
DOI:10.1007/s00347-020-01177-4.

C. Schmidt, B. Grundel, H. Schumann, and P. Rosenthal. "Annotations in Different Steps of Visual Analytics". In: *Proceedings of VISIGRAPP.* 2021, pp. 155-163.
DOI: 10.5220/0010198001550163.

Kurzfassung

Annotationen sind nach dem Verständnis in dieser Arbeit Informationen die zusätzlich in ein Visual Analytics System eingebracht werden. In der Literatur gibt es hierzu bereits zahlreiche Beispiele. Viele dieser Beispiele zeigen die konkrete Umsetzung von Annotationen, die in einem speziellen Anwendungsfall unterstützend wirken. Eine systematische Analyse der Annotationscharakteristiken hingegen ist nach wie vor Gegenstand der Forschung und eine große Herausforderung. Mit Hilfe einer solchen Analyse wäre es möglich, angemessene Charakteristiken sowie ein passendes Design für Annotationen systematisch zu entwickeln, also ein *Annotationsproblem* zu lösen. Wir adressieren die systematische Lösung von Annotationsproblemen, indem wir zunächst die Charakteristiken vorhandener Annotationslösungen herausarbeiten und anhand grundlegender Fragen zu Annotationscharakteristiken in einer Morphologischen Annotationsbox strukturieren. Darüber hinaus entwickeln wir ein Modell, welches die Morphologische Annotationsbox für die weitere Analyse zugrunde legt. Unter Heranziehung der Anforderungen aus dem Anwendungsfall sowie aus der Visuellen Datenanalyse können mit Hilfe des Modells angemessene Annotationscharakteristiken ermittelt werden. Hierdurch müssen die Annoationsmöglichkeiten im konkreten Anwendungsfall nicht mehr aufwändig aus allen Möglichkeiten herauskristallisiert werden, sondern werden mit Hilfe des Modells systematisch entwickelt. Im weiteren Verlauf zeigen wir, wie man ein passendes Design für die Annotationen mit den angemessenen Annotationscharakteristiken entwickeln kann. Dieses Design steht dabei in direkter Beziehung zur Visualisierung der Originaldaten und erfüllt die Anforderungen sowohl der Originaldatenvisualisierung als auch die Anforderungen der Annotationsvisualisierung. Um die Anwendbarkeit nachzuweisen, implementieren wir das Annotationsdesign in ein bestehendes Visual Analyticssystem. Mehrere Sitzungen zur Erlangung von User Feedback und die testweise Anwendung des Modells auf epidemiologische Daten zeigen, dass diese Annotationsermittlung zu einer besseren Datenvorbereitung, Datenbereinigung und Exploration von heterogenen klinischen Daten beitragen kann.

Abstract

Annotations in this work are understood as supplementary information integrated into a visual analytics system. Various examples in literature show their usefulness and application. However, many of these examples contain a specific annotation solution to support the analysis of a particular use case. A systematic approach to analyze and develop the annotation characteristics is still ongoing research and difficult to address. Such an analysis could help to systematically characterize and design annotations in accordance with the needs of a particular use case, in short: to solve an *annotation problem*. We address the development of a systematic solution for the annotation problem by conducting a literature survey on existing annotations and extracting the annotation characteristics. We structure these characteristics based on basic questions on annotations and sort them into a morphological box. Furthermore, we develop an annotation characteristics model, which takes the morphological box as a base. The model additionally takes the requirements from both the use case and visual analytics into account and derives suitable annotation characteristics for this particular use case. This helps domain experts and visual analytics experts to systematically develop suitable annotations. Additionally, we show how a fitting design for the annotations can be developed for the derived suitable annotations. This design is aligned with the original data visualization and fulfills both the requirements from the original visualization and annotation visualization. To show the usability we implement the annotations into a visual analytics system for heterogeneous clinical data. Several user feedback sessions and a brief approach to apply our model to epidemiological data show that our approach can help to improve data preprocessing, data cleansing and data exploration for clinical data.

Contents

List of Figures

1 Introduction

Within the last decades, clinics have collected tremendous amounts of data. These data come from clinical management systems, medical apparatuses, and manual personnel recordings. Over the years, clinical management systems have been updated, new data fields were introduced, others changed or removed, if deprecated. Medical apparatuses have been improved allowing the use of additional sensors, adding more or preciser data. On top, physicians and other personnel come and go, with each person carrying his or her own data recording philosophy. All of these factors lead to complex data with many peculiarities. Nonetheless, the data are of high value, as they may contain hidden information, precious to physicians as the information may lead to improved diagnostic or therapy decisions.

Visual analytics has proven to be a powerful method to extract that valuable information from data. With suitable techniques, such as data preprocessing [Fam+97], which allows a structuring and consolidation of the data, or data cleansing [Gsc+14], which allows data editing operations to improve the data quality, visual analytics can tackle these peculiarities. With these steps described above, structured, and corrected data sets can be generated for a thorough data exploration.

Nonetheless, the data alone may not be sufficient. Additional information, such as how the data should be structured for the analysis, which data values have been edited, or what hidden information has been explored can be necessary or even critical for the analysis process [Lip+10; MST12]. To integrate that additional information, annotations come into view. Annotations are used to enrich the visual analytics system with additional information [HS12]. This information is brought into the system via gathering processes and then communicated to the user, where needed. In doing so, annotations can support the structuring, consolidation, editing, and exploration processes.

1.1 Motivation

There are many examples of annotation usage in the literature. Most approaches address particular issues on an individual annotation problem, providing solutions for a specific use case. A prominent example is the usage of annotations to integrate the “implicit error” into the analysis. During the zika virus outbreak in South and Middle America, data on the outbreak were collected in several countries. McCurdy et al. [MGM18] developed a visual analytics tool to present the collected data. As

the data collection methodology varied between countries, a so called “implicit error” occurred. Without correction, this error could lead to misinterpretation of infection numbers, as they were not fully comparable between countries.

In their work, McCurdy et al. gave experts the possibility to integrate their knowledge on these errors via annotations into an outbreak visualization system, helping others to understand them and so better interpret the zika virus data. A view

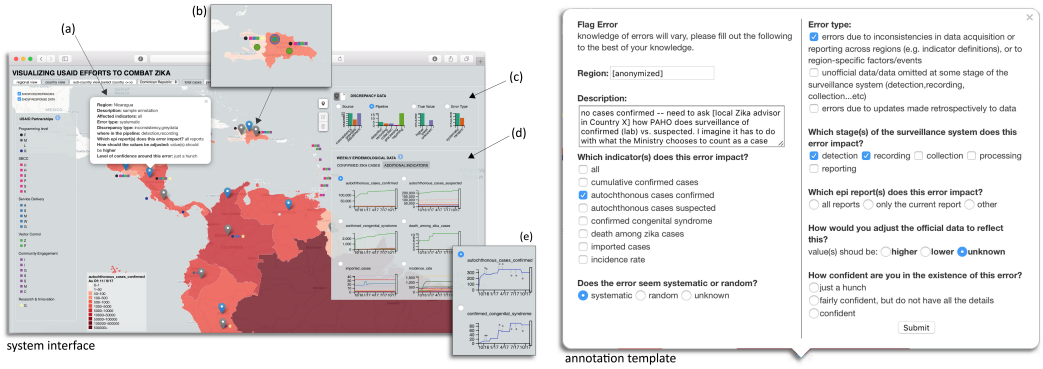

Figure 1.1: The annotation example, taken from McCurdy et al. [MGM18]. (Left) The visual analytics system is enriched with additional information, which we call annotations. The information can be shown on top of the visualization (a), encoded in the data (b), or in a separate view (c), (d), & (e). (Right) The annotation recording in this case is done via a template which records categorical information and free text.

on the visual analytics system and annotation approach can be seen in Figure 1.1, which stems from McCurdy et al. [MGM18]. The use of annotations works conveniently in that approach yet leaves open on why the particular characteristics for the annotations were chosen. For example, it would be interesting, why they decided to encode parts of the annotation information directly into the data (b). Encoding the unverified annotation information directly into the data visualization could lead to additional interpretation errors if the annotation information is incorrect. Furthermore, McCurdy et al. decided to gather the annotation information view an annotation template, which is a common way to take user input (Right side of Figure 1.1). Another suitable way to record information could be a screenshot of the visualized zika data with markings and/or comments from experts.

The questions mentioned above apply to many annotation applications in literature (elaborated in our literature survey in Chapter 2), so that the actual process, of how to derive annotation characteristics within an overarching model is ongoing research. Such model could provide methods to allow a systematic generation of ways to gather annotations from and communicate annotations to domain experts, in order to improve the overall analysis. This could not only improve the use of suitable annotations, but also systematically explain the choices for annota-

tion characteristics for readers. Additionally, the model could collect a set of key annotation characteristics, available for approaches to be designed.

1.2 Challenges and Goals

The development of an overarching model bears several complications. First, the term "annotation" is frequently used in literature, but a suitable definition is hard to find. In general, annotations are often associated with comments made by users of computer systems. There are examples in visual analytics, where annotations also contain other explanatory information or categorization, making annotations multi-dimensional. For clarity, the term "annotation" in this work describes more specifically information within the visual analytics system, which is added during the visual analytics process.

Second, the integration of annotations in the field of visual analytics is often done on top of a data visualization. Since it is complementary information, there is existing information that already fills the available screen space for visualization and interaction. Consequently, a conceptual annotation model needs to provide methods which enable the development of gathering and communication techniques with respect to the underlying original data and visualization. Even though this is done in close relationship to the respective use case, a generic approach, able to apply to various use cases, needs to resort to a set of key annotation characteristics. The development of such a set is not trivial, as the set should provide characteristics able to tackle the vast variety of data and visualization possibilities. Naturally, such a set can hardly be complete.

Furthermore, the limited screen space and interaction methods available for data visualization and annotation visualization leads to a concurrency situation. For example, if a particular screen space is used for data visualization, it is not available for annotation visualization at the time and vice versa. A compromise is needed. Applying the conceptual model should provide a solution with such a compromise for specific visualization and annotation situations.

To solve these problems, this work has the goal to derive an annotation characteristics scheme, which collects and sorts annotation characteristics found in the literature into a set of *key characteristics*. With that scheme at hand, we have the goal to design an annotation characteristics model which is able to systematically design annotations for a particular use case. With that model it is possible to reduce the set of *key characteristics* into *suitable characteristics*. Suitable characteristics are a subset of the key characteristics and specifically suitable for a particular use case. These suitable characteristic need to be implemented with a dedicated design into a visual analytics system either from scratch or into an existing one. Altogether, we consider this an *annotation problem*.

1.3 Approach and Contribution

The questions which arose in the previous section become more urgent, as the number of publications with annotation integration increases and the visualization community recognizes the growing importance of annotations. To contribute to answering the questions, we move towards a structuring of the creation and usage of annotations, which should ease the development of visual analytics systems with annotation. We reach this, by (i) developing an annotation characteristics model, (ii) deriving suitable annotations for common steps in visual analytics and (iii) applying our approach in cooperation with experts on a use case.

1.3.1 Annotation Characteristics and Conceptual Model

As the definition, design, and usage of annotations moves forward, we also integrate different means of additional information into the system. We analyzed and published the means and impacts of that additional information in combination with the data ([Röh+19a], [Röh+19b], and [Röh+19c]). We showed in our work that the combination of data and additional information, either automatically generated by the system or externally integrated can be beneficial for the analysis.

A further prerequisite for finding annotation characteristics is to identify them in visual analytics literature. We conduct a literature analysis resulting in a comprehensive description of characteristics. During this process we identified four basic questions about annotations: "What are annotations?," "Why annotate?," "How to gather annotations?," and "How to communicate annotations?." To allow annotation developers to set up their annotations' specifics, they need to choose and combine suitable characteristics from the existent characteristics and for every basic question. This is supported by the principle of creating a morphological box (see Zwicky [Zwi67] for further information). We establish such a morphological box which lists the four basic questions and sorts the existent annotation characteristics into them, to define the key characteristics. The idea of morphing various possibilities of input has been examined in advance through the example of data encoding variants for retinal data. We presented our approach at the *EuroVis Workshop on Visual Analytics (EuroVA), 2017* [Röh+17].

With the morphological box we have a structured annotation characteristics overview (the *key characteristics*) at hand, from which individual annotations can be created. We have presented this approach at the *9th International Conference on Information Visualization Theory and Applications (IVAPP)* [SRS18].

Our morphological box forms the basis for our conceptual annotation model. While the morphological box shows all possible combinations of key characteristics, not all of them are useful for a specific use case. For example, free text annotations, manually recorded by user, may not be sufficient for the automatic processing of millions of data points. So, we develop a conceptual model which organizes the actions

necessary to develop use-case-tailored annotations. It ensures that the correct prerequisites are met via an "Assessment of the Requirements." Suitable annotation characteristics are identified by "Reducing the Morphological Box" and matched to visualization/interaction techniques are found via "Designing the Annotations." With the help of the model, a systematic enrichment of visual analytics approaches with annotations is possible.

1.3.2 Annotation Development for Visual Analytics

For the challenge of systematically integrating annotations into visual analytics, we first examine existing steps in the analysis. Here, data preprocessing, data cleansing, and data exploration are common [Gsc+12; Sac+14]. Each of these steps has its own challenges. We present a sophisticated analysis of the advantages of combining these steps in a user-in-the-loop workflow with the help of annotations in our work on *Combining Visual Cleansing and Exploration for Clinical Data* presented at the *IEEE Workshop on Visual Analytics in Healthcare (VAHC), 2019* [Sch+19].

Data preprocessing often requires consideration of multiple data sources, which can lead to redundant and potentially conflicting data point values. In our article in *The Visual Computer, 2018* about *"Visual Analysis of Retinal Changes with Optical Coherence Tomography"* [Röh+18], we pick up the aspect of unifying different data sources prior to the analysis.

During data cleansing, the detected data discrepancies and incompleteness must be resolved to create a consistent data set. During data exploration, the data must be assessed by experts with domain knowledge to identify findings that may lead to new insights. Annotations in these cases can contribute by (i) marking and communicating data redundancies and discrepancies, (ii) informing users about data cleansing decisions or recurring data errors, and (iii) perpetuating and/or allowing comments on results in single-user, asynchronous, or collaborative environments. A basic analysis was presented in our work on *Varying Annotations in the Steps of the Visual Analysis* which we published in a technical report in the *ArXiv Information Vol. 10, No. 9, 2019* [SRS20].

This thesis addresses these issues by designing tailored annotations with the help of our conceptual model for each of these different steps. For the data preprocessing step, we insert automatically generated annotations for data value redundancy, discrepancy, and discrepancy resolution. For data cleansing, we integrate annotations that enable users to explain decisions about resolved discrepancies and to automatically detect recurring errors on the other hand. The latter facilitates the further cleansing process by reducing the effort for the detection of recurring errors. The annotations for the exploration step are designed to capture the users' knowledge, required for the analysis, to support identification and externalization of findings and insights, and allow for user communication. Our annotations follow the principle of being as automatic as possible, while also increasing the trust in the data by

reliability and transparency. Under this premise, we identify and describe these customized annotations for individual steps in the visual analytics process, generating an annotation concept for these steps. This work was presented at the *12th International Conference on Information Visualization Theory and Applications (IVAPP)* with the title *"Annotations in different steps of visual analytics"* [Sch+21]. We are aware of the fact that this problem also affects other steps of the analysis, such as validation or knowledge generation. However, integrating annotations into these steps requires further detailed considerations, and goes beyond the scope of this work.

1.3.3 Application on a Use Case with Expert Feedback

To this end, we developed a generic model and solution for annotation creation in visual analytics. To test their applicability to real-world visual-analytics problems, we used clinical data from the ophthalmological domain. We conducted the annotation development process with a requirements engineering step for visualization and annotation. We have found that the structuring and consolidation of heterogeneous, redundant, and conflicting clinical data is easier to overlook and understand when physicians are informed about the consolidation process. This information is made available by automatically collecting annotations during data structuring and consolidation and communicating them in a way that ensures the changes are transparently reflected without distorting the original data. Furthermore, we automatically structured and annotated free text provided in doctoral letters via a text mining approach. In doing so, certain values, such as the visual acuity values, can be extracted, structured, and saved via annotations. This approach was published in the journal *"Der Ophthalmologe," 2020* with the title *"Merkmalsextraktion aus klinischen Routinedaten mittels Text-Mining"* [Gru+20]. During data cleansing, we still managed to keep track of the data in the tightly timed and interrupted clinical environment by using the model to design annotations that show who made what changes, when, and with what results. This puts clinicians in a position to identify changes, assess them, as well as pick up where they left off. Furthermore, it opened the possibility for the experts to delegate editing tasks and limit themselves to the control function. Finally, it managed to capture the results of the analysis during the data exploration in such a way that they were able to work with techniques beyond our tool using externalizations, for example, to create presentations and conduct further research using familiar tools. This work was presented at the *"12th workshop on Visual Analytics in Healthcare (VAHC) in conjunction with IEEE VIS 2021"* [Sch+19].

1.4 Thesis Structure

The explanation of the research conducted in this thesis continues as follows: To better understand the background of this work, we provide a short introduction into

the domain of ophthalmology in clinical environments at the beginning of Chapter 2. Here, important terms from the medical field are explained. Afterwards, the current literature is examined with regard to supporting annotations and gaps are pointed out. Subsequently, in Chapter 3, a model, a morphological box, and a user-in-the-loop workflow are established. They address the gaps by classifying annotations in terms of their characteristics, providing modes to find suitable annotations, and relating them to common steps in visual analytics. Chapter 4 describes suitable design possibilities for collecting and communicating the annotations in relation to the annotation characteristics model. Chapter 5 implements this design to our use case in the medical environment and provides a short glimpse into the use of our approach on epidemiological data. The use case analyzes ophthalmic data from one clinic, while the epidemiological data concerns SARS-CoV2 data from several clinics. Chapter 6 reflects the feedback from the domain experts on the application and implementation for heterogeneous medical data. Chapter 7 draws a conclusion from the considerations by discussing the benefits of the morphological box, the model, and the user-in-the-loop workflow. The question is taken up to what extent the annotation model achieves the goals of increasing transparency, documentation function as well as externalization of results. Furthermore, future research questions are outlined.

2 Fundamentals and Related Work

Chapter 2 intends to provide information on the use case and the state of the art for annotations in visual analytics, as these represent the fundamentals of this work. In the first part, we describe the medical background and environment for a better understanding of the creation of our annotation model. The second part outlines current annotation solutions and situates them within the scope of an overview on existent annotation characteristics, including the identification of open issues.

2.1 Use Case

Throughout the thesis, we will refer to a typical use case that helps us to demonstrate the issues raised with annotations. With its description hereafter, we allow for later understanding of the process that led to characterizing annotations, developing the conceptual model, and applying it to actual data.

The use case described is situated in the medical domain. Physicians in clinical environments work on retinal diseases by examining patients and thereby collecting large amounts of electronically available data. In the following we provide a short introduction into the domain and describe the data peculiarities.

2.1.1 Medical Background

Visual acuity is the ability to see one's environment sharply and in detail. It highly depends on the condition of the macula, which is the central part of the retina within the human eye. One of the most common macula impairments is age-related macular degeneration (AMD), which is an important cause of blindness in industrialized countries [Muñ+00]. A significant body of research has been devoted to find treatments for this ailment, resulting in successful therapy methods like the injection of anti-VEGF (vascular endothelial growth factor) agents directly into the affected eye. Through this therapy, many patients are able to sustain or even improve their visual acuity [BLM12]. Short term effects as well as long term impact have been proven beneficial [Adr+18; NS10]. Yet, to date different facets of the dependencies between injection frequency, medication and the visual acuity development are not fully understood, especially with respect to real-world patient data. Even though more injections over longer time periods can have positive impact [Adr+18], each injection bears the risk of infection as surveyed by Falavarjani and Nguyen [FN13].

Weighing the tradeoff between potential positive impact and infection risk imposes a heavy responsibility on retinal physicians. Providing meaningful information by means of data analysis of existing patients' developments could effectively support such decision making. This is particularly important, as there are often many extraordinary incidents in a patient's medical history, like cataract surgery or pigment epithelial detachment, which may distort the overall development of visual acuity.

2.1.2 Data Characterization

Our domain experts[1] work in a clinical environment where most data are gathered and stored electronically. To handle the clinical data, which are management data, device data, examination data, and treatment data, a hybrid system consisting of different commercial tools and an in-house web application exists. The *management data* are recorded by clinic assistant personnel via a hospital information system developed in-house. The management data mainly contain general information, like age and sex. The eye-related *device data* are recorded by the respective device software, including image data, meta data, and abstract data like visual acuity values and tension values. The *examination data* are recorded via proprietary digital forms, created in accordance with analog forms that were previously used. During an examination, an assistant or a retinal physician fills in the collected information. All data are summarized in a semi-structured medical report letter, which is manually filled in by the retinal physician and completed with *treatment data* as well as free text comments from the retinal physician. Treatment data are additionally generated by the surgery management system that is used to plan and document operations, such as anti-VEGF injections or cataract surgery. The data are organized by creating one record for each appointment per patient. Ophthalmologists select relevant patients in their clinical system and extract anonymized records. As a prerequisite, each selected patient needs to have received at least one injection of anti-VEGF medication, assuring that treatment in some form has been applied. Altogether, we obtain data from records of more than 3,500 patients between the years 2004 to 2018 spread over 95,000 files stemming from different applications and devices in the clinic. The patients have between one and 131 appointments, distributed over a time span between one day and 13 years. The content of the medical report letters for each appointment is partly extracted into structured data files by a text mining approach.

To increase the level of certainty, there are deliberately included redundancies in the data, as some data points are redundantly recorded by different devices or personnel. On the other hand, there are also incomplete data points, as some of the clinical devices may not have been available over the full period of time. Generally, the data were initially gathered for clinical use.

[1] Ophthalmologists who work in an eye care center with the specialty on retinal diseases.

2.1.3 Domain Task Support

To use the outlined data for research, we have to cope with large amounts of abstract, heterogeneous, incomplete, and redundant data. Hereupon, ophthalmologists aim at performing a variety of tasks. For instance, they first want to get an overview on a single patient's situation, implicating the display of general patient data. This requires the availability and consolidation of that patient's data. Annotations in this case can help to provide feedback for the consolidation process, including possible existing comments from other ophthalmologists. This would complement the users' view on the patient.

A second task is the comparison of two or more patients' developments over time. Annotations can support the reasoning process in this case if they provide means to mark and preserve the findings in such a comparison process. An example is the drawing of connection lines within the visualization to indicate similarity.

As a result of the comparison process, a third task evolves, as physicians want to sort the patients in groups of similar visual acuity development. In this case classification annotations, for example, can be of use.

2.2 Annotations as a Means to Integrate Additional Information

Annotations have been widely used to integrate additional knowledge into the analysis. Existing literature depicts annotation principles, describes annotation usage examples, or works with previously recorded annotations. The following sub-sections outline the relevant state of the art and show the gaps addressed in this work.

Generally speaking, annotations have been recognized as important or even critical for various fields of research. The linguistic field is one of the early domains where annotations have been examined. That domain identified the need for annotations to be gathered and applied. Furthermore, strict rules are needed for annotations to be beneficial in the field of linguistics [Fro17; IR04]. Also, the quality of annotations has been examined, e.g., Snow et al. [Sno+08] find that four non-expert annotators are needed to reach accuracy of one expert in the field of linguistic annotation.

In visual analytics annotations have also been taken into account for several years, as the use of annotations has been found critical [Lip+10; MST12] and plays a role in different perspectives. By using a video annotation system, for example, an experienced sports analyst can generate more accurate event data during or after a game [CG16]. Additionally, existing annotation techniques have been described by Heer and Shneiderman [HS12] and mentioned as a suitable technique in the knowledge generation model of Sacha et al. [Sac+14]. For provenance examination, annotations are seen as substantial by Ragan et al. [Rag+16]. Furthermore, annota-

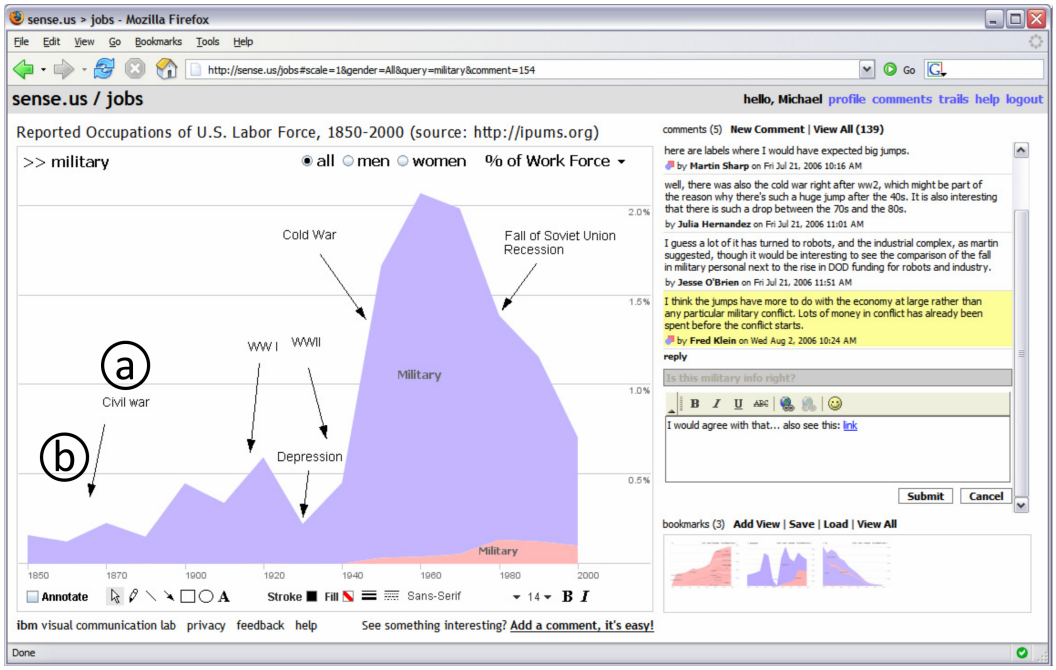

Figure 2.1: This illustration from Heer et al. [HVW07] shows different examples of existent annotation characteristics. There are free text annotations (a) as well as graphical items (c).

tions themselves have been the subject of investigation, as seen in the classification of annotations by [Van+18].

While these approaches show the importance of annotations and bear some promising ideas on annotation characteristics, we aim at providing a general point of view. As a prerequisite on the detailed characteristics and model descriptions in the following chapters, we examine 32 existing annotation examples. We have structured our minor survey with respect to our classification of annotation characteristics to ease the understanding of our mental map and we classify the related work into four basic questions: (i) "What are annotations?," (ii) "Why annotate?," (iii) "How to gather annotations?," and (iv) "How communicate annotations?."

2.2.1 What Are Annotations?

Among the 32 examples in literature, we identified four different key annotation characteristics for "What are annotations?." *Categories*, for example, are prominently used to either manually or automatically annotate images. In this case, class labels are created to train a machine learning algorithm to classify images [Cha+03; WH11]. A more implicit way to use category annotations is supported by re-CAPTCHA. While first CAPTCHAs (Completely Automated Public Turing test

to tell Computers and Humans Apart) only had the purpose to prevent massive bot abuse on websites [Ahn+03], reCAPTCHA is nowadays used to include internet users in annotating images [Ahn+08]. Other broad fields for category annotations are genome biology, where genome sequences are annotated [Con+05], or the linguistic domain, where Snow et al. conducted studies, in which experts and non-experts manually categorize texts [Sno+08]. Category annotations for classification are also applied in other contexts, such as economic data [SW15]. Furthermore, category annotations can also apply to scenarios, where structured input from users is needed. This is, for example, the case in implicit error correction, described by McCurdy et al. [MGM18]. Here the user may choose from different categories to add information to the system. This method is used to categorize the user's input for further internal processing as also proposed in other work [MT14; Wil+11; Zha+17].

In some of the aforementioned publications ([MT14; MGM18; Wil+11]) the categorization is used to sort *free text* comments, which we have identified as the second characteristic of annotations. Here, free text annotations are used to integrate additional information in the words of the user with few or no restrictions. Another case is the work from Alm et al. [AAU15], where additional information on industrial devices is integrated to ease the maintenance process. The free text here is used to enrich and explain complex maintenance tasks. Free text annotations are also used for communication with the analysis system. This is to input or answer analysis tasks or questions [EB12; GS06; HVW07; Wil+11]. In some cases, free text annotations were taken offline (as notes), e.g., by Mahyar et al. [MST12] or Lipfort et al. [Lip+10]. Yet, these examples and others ([Bou+17]) noticed that annotations should be persistently made available within the system.

While the first two characteristics of annotations are text-based, we have identified *graphical items* as another characteristic of annotations in the literature. They are often used to highlight a finding or insight in the visualization [EB12; GS06; HVW07; Wil+11] or to indicate the position of an annotation content [MGM18]. Other purposes to use graphical items are to explain images in the case of Alm et al. [AAU15], the encoding of graph-based annotations in Mahyar et al. [MT14], or the encoding of automatic annotation content of Klien et al. [KL05].

The final characteristic of annotations on "What are annotations" is the collection of *provenance information*. We define provenance information as information about the performed visual analytics process. We see it as a characteristic on its own because it is a frequently used annotation type, and has a complex structure, which cannot be reduced to one of the other characteristics without losing parts or all of its semantic content. One example is the automated recording of interaction events during the analysis.

The importance of provenance annotations was outlined by Ragan et al. [Rag+16]. They see them as a method to describe interaction provenance and recording of findings and insights. Another reason to use provenance annotations is to sup-

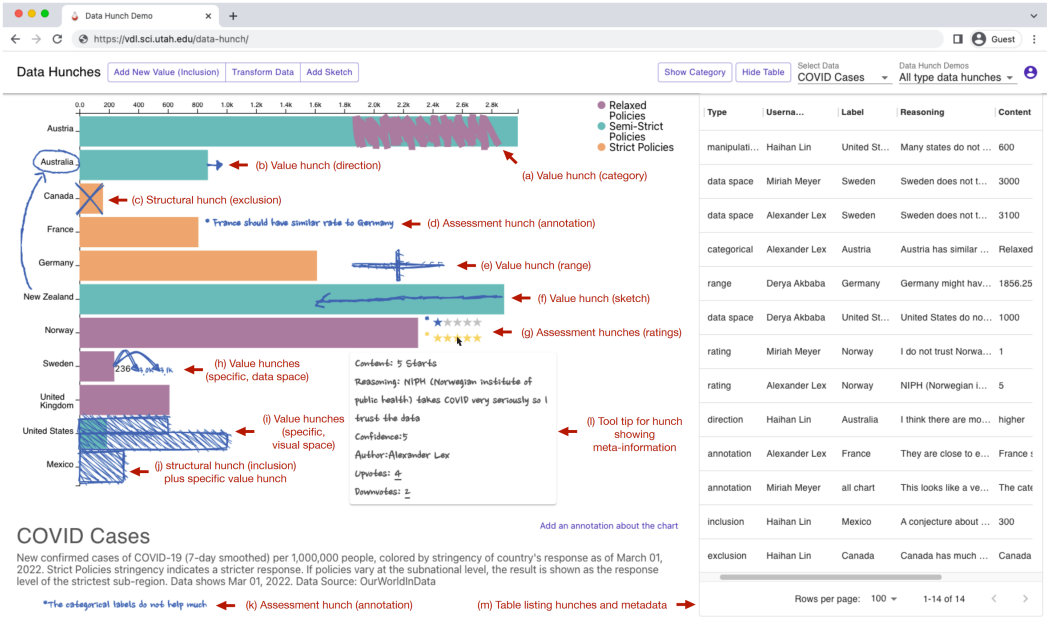

Figure 2.2: Lin et al. [Lin+23] use annotations to amend a data visualization with user information. The letters (a) to (m) in red color explain the visual representation and semantic the authors used for their annotations.

port collaboration between experts, as proposed by Zhao et al. [Zha+17; Zha+18], who present a dynamic graph visualization, which enables meta-analysis of data based on user-authored annotations. In a similar way Al-Naser et al. [Al-+13] combine the storage of the raw data with the storage of the interpretations produced by the visualization of features by multiple user sessions. They reproduce users' amendments to the interpretations of others and the enable to retrace the history of amendments to a visual feature.

Finally, provenance annotations can assist experts in the recalling process of the analysis [Bra+14; Cal+06]. In this way, provenance information (e.g., what exploration actions are performed) is recorded and communicated within the visualization system to better understand previous sessions.

2.2.2 Why Do We Annotate?

There are various reasons to add additional information to an analysis. We grouped these reasons into three categories, which define the three key characteristics for our second basic question: (i) add data information, (ii) add user information, and (iii) add outcome information.

One reason to add *data information* is to explain the correction of erroneous data as McCurdy et al. [MGM18] have done with "implicit errors." Also, Boukhelifa et

al. [Bou+17], who enable participants to assigned notes to data to collect information on data. Similar, yet with another intention, is the integration of annotations for data assessment, e.g., on data quality, redundancy, discrepancy, etc.. Here, not the explanation of data corrections, but the assessment of existing data is focus, as Boukhelifa et al. [Bou+17] have shown. Adding structuring information is another form to add data information, as Röhlig et al. [Röh+18] performed in unifying data from different electronic devices. Klien et al. [KL05] even produce new data from the semantics of data annotations, as they integrate the information into an ontology from which a classification for existing geospatial data is derived. By that, they can show a semantically enriched landscape.

The second purpose for annotations is to add *user information*. In the case of additional information in connection to data, it is important to distinguish between user information and the previously described data information. While data information directly contains data related facts (e.g., "What is wrong?"), user information contains knowledge from the user about the use case and not the data ("Why is it wrong?"). Good examples are McCurdy et al. [MGM18], who dedicate a separate window, to both integrate data and user information and Boukhelifa et al. [Bou+17] who also provide users' assessment of data quality. Recently, Lin et al. introduced data hunches, where they allow users to integrate their knowledge into a visualization tool [Lin+23]. An example of their tool is shown in Figure 2.2.

Another common application of annotations is the integration of user knowledge to classify images [Cha+03; Ahn+08], genome sequences [Con+05], or linguistic elements [Sno+08]. Collaboration also plays a major role, for example, when the knowledge of experienced workers in industry is integrated into maintenance systems to help less trained personnel [AAU15]. Other collaborative applications are the identification of points/areas of interest in visualizations [GS06] or users' interpretation of geo-spacial data [Al-+13]. A feature-driven approach is depicted by Heer et al. [HVW07], who amend user knowledge by integrating background information for specific features within the visualization for explanation. A more implicit way to integrate user information is the recording of interaction information from users, as done by Bradel et al. [Bra+14], Callahan et al. [Cal+06], and Ragan et al. [Rag+16].

Our third reason to annotate is the amendment of *outcome information*, which we understand as new information resulting from the analysis of the data. It appears to be an important reason in literature. Often, findings in the data are identified and preserved by an annotation. Examples are the users' selection and description of geological peculiarities in spacial data by Al-Naser et al. [Al-+13] or the general possibility to describe and preserve findings and insights [EB12; HVW07; Lip+10; MST12; MT14; Wil+11]. One step further are the approaches by Zhao et al. [Zha+17; Zha+18], who provide "annotations on annotations" within an extra tool, to allow collaborative analysis of the externalized outcome information.

We also consider outcome information as the recorded results of classification algorithms, where semantic labels are assigned [Cha+03; Con+05; SW15; WH11].

2.2.3 How to Gather Annotations?

While the first two aspects of characterization concerned the annotation content, the question "How to gather annotations?" relates to their formal characteristic. In our literature excerpt we found five key characteristics to gather annotations.

Very common is the plain intake of *alphanumerical input*. More than half of our use cases provide such methods by allowing users to type in their information (see Figure 2.6). The recording is mainly done via direct input into the system [AAU15; Bou+17; EB12; GS06; HVW07; KL05; MT14; MGM18; Sno+08; Wil+11; Zha+17; Zha+18], while two approaches [Lip+10; MST12] used offline commenting in their studies.

In contrast to alphanumerical input, placing *marks* resembles a low-level way to gather annotations. Here, often specific features or findings can be identified and preserved within the visualization by setting a mark, such as circles, arrows, or other glyphs. All of our literature examples, who use that technique [EB12; GS06; HVW07; Wil+11; Zha+17], combine this method with other gathering techniques, e.g., alphanumerical input.

The third way to gather annotations is *selection and brushing*. This includes data points to be selected for annotation, which was performed by various approaches [Al-+13; Bou+17; HVW07; MGM18; Zha+18]. Additionally, specific pre-defined annotations can be selected from a list, to allow structured annotation creation [MST12; WH11; Wil+11; Zha+17].

A direct way to preserve a visualization is the recording of *screenshots* for annotation. This can be done during the visual analysis, as Groth and Streefkerk [GS06], Willett et al. [Wil+11], as well as Heer et al. [HVW07] have shown. Alm et al. [AAU15] on the other hand, take their screenshots during maintenance tasks in industry.

Finally, *automatic computation* is a gathering technique, where additional information is automatically recorded by the respective analysis system. The annotations can be created from an ontology or pre-defined structures [AAU15; Röh+18], from interaction parameters or provenance information [Bra+14; Cal+06; Rag+16], from algorithms or maxima/minima calculations [Cha+03; Con+05; EB12; SW15; WH11], or from previously recorded manual queries, transformed into automatic data selection and annotation [KL05].

2.2.4 How to Communicate Annotations?

Not all approaches on annotations are readily designed to communicate the gathered annotations [Bou+17; Bra+14; Rag+16; RSS16; Sno+08; Ahn+08; WH11]. This is also expressed in the approaches from Mahyar and Tory [MT14] and Zhao et al. [Zha+17; Zha+18], who take annotations from other applications for a dedicated annotation analysis tool. Some of these tools also represent solutions, where more than one communication technique is used. In sum, we identified three different techniques.

The *visual separation technique* provides a distinct space, often an extra view for the presentation of annotations to the user. This view is either located next to the visualization in a multi view environment [AAU15; EB12; HVW07; Wil+11] or opened on demand as a separate window [Cal+06; MGM18]. While not electronically, yet we can consider offline annotations [Lip+10; MST12], such as notes taken on paper also as visual separation, as they do not interfere with the actual visualization. The aforementioned dedicated annotation analysis tools [MT14; Zha+17; Zha+18] also bear separate views, e.g., comment sections.

From these approaches [MT14] and [Zha+18] also represent *visual encoding*, which we consider as the combined encoding of data and annotations in the same view with the same techniques. Further examples are the use of color-coding annotated data within the original view, as done by Al-Naser et al. [Al-+13], Shabana and Wilson [SW15] as well as Willett et al. [Wil+11]. Fully integrated annotation encodings come into view, where classification annotations are used, and the results are presented by visualizing the data classes or aggregated forms [Cha+03; Con+05; KL05].

As a contrast, *layered visualization* represents a technique to show annotations on top of the data visualization, but in a separate view with distinct visualization techniques. This often concerns highlighting functions, e.g., of findings in the data [EB12; GS06; HVW07] or the communication of explanatory annotations, e.g., with tooltips [SW15; Wil+11]. Differently, layered visualization can also be used to place glyphs as local indicators for more complex annotations, which are shown on interaction with the glyph [MGM18].

2.3 Annotations in Visual Analytics

Having explored annotation literature in principle in the last section, yet already with a focus on the field of visual analytics, we now want to classify them into different phases of visual analytics.

Of the various steps in visual analytics, we have dedicated ourselves specifically to the steps data preprocessing, data cleansing, and data exploration. The reason is that first, previous analyses in the field of heterogeneous real-world data have

shown their importance [Gsc+12], and second, the steps were highly applicable to our domain with its heterogeneous and erroneous data.

As data preprocessing generally has the goal to structure and fuse the data, *data preprocessing annotations* support that process by gathering additional information. Existing literature shows automatic [Jin+17; Lak+18; SW15] and manual [Krü+15; Sch+19] approaches. For the communication of these annotations, Krueger et al. [Krü+15] have shown that a direct communication within the data visualization can be useful, while Shabana and Wilson [SW15] communicate the added information as an extra layer on demand. Although there are approaches to combine both direct and on-demand communication [Sch+19], a thorough analysis of such presentations is ongoing research.

The reason for data cleansing is the correction of erroneous data [MF05]. *Data Cleansing Annotations* can support this process when they integrate the knowledge of the user. McCurdy et al. [MGM18] apply this approach to epidemiological data, where they gather the information from the user via an extra view and communicate the information on demand via interaction functions in the visualization system. While there are further approaches for data cleansing visualizations [Gsc+14; Sch+19], we focus on annotation use for recording and visualizing the circumstances of the cleansing process.

Data Exploration Annotations have been used to support the exploration step by, e.g., (i) localizing the findings [HVW07; Wil+11], (ii) documenting the findings [Wil+11; Zha+18], and (iii) externalizing the findings and, if applicable, the gained insights [Zha+17]. Data exploration annotations can be gathered either directly in the visualization [GS06], next to the visualization [Wil+11], or via extra views [Sch+19]. Concerning the communication of annotations during exploration, Groth and Streefkerk [GS06] and Heer et al. [HVW07], among others, show them directly in the visualization, while Zhao et al. [Zha+17] and Mahyar et al. [MT14] design a dedicated tool for annotation visualization.

2.4 Discussion

For the literature analysis presented in the previous sections, 32 suitable publications were identified and subjected to a closer examination. Identification of relevant literature for description, classification, and use of annotations was performed using a combination of keyword search and content search. The keyword research was done in several online publication libraries, such as IEEE Xplore, ACM Digital Library, Elsevier, and google scholar. The content search was performed for publications presented at well-known conferences of the visualization community, such as VIS, EuroVis, PacificVis, CHI, and VISIGRAPP for the years 2015 to 2023. It was found that annotations are used in visual analytics and beyond.

First, we have analyzed the publications in respect to general properties, such as the annotation topic of the paper and the analysis steps addressed. Figure 2.3 shows the assignment of the individual publications in this respect.

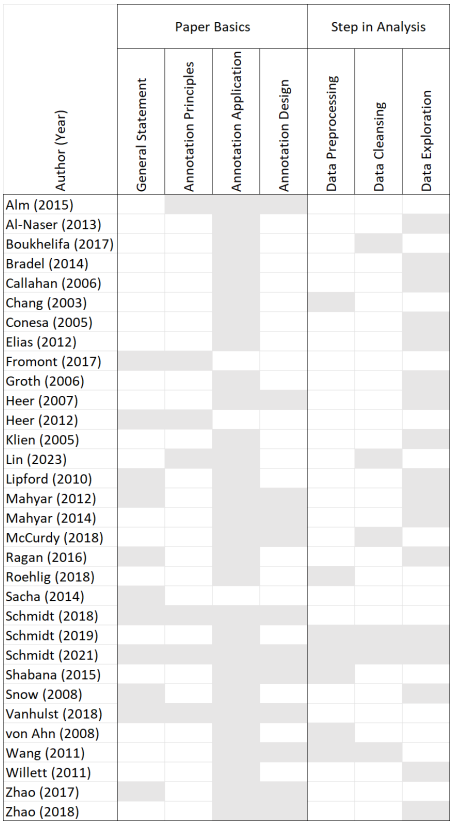

Figure 2.3: An overview on the general scope of the publications analyzed in this work. The paper basics (left part of the table) describe the main premise of each publication. The right side shows the affected steps in visual analytics on paper level. (grey = affected in paper)

In elaborating the topics, we were able to identify differences in the basic treatment of annotations. This concerns the assignment of literature to certain main topics in the annotation environment as well as to individual steps within the visual data analysis (Figure 2.4, and Figure 2.5). It turns out that in most cases annotations are specifically applied to a use case, e.g., to support a use case analysis, while annotation principles, general statements, or papers dedicated to the annotation design are less addressed. To our knowledge, there is no basic systematic modeling and characterization. Concerning papers with annotations within the visual analytics field, the accumulated result reveals that the absolute majority of the analyzed publications addresses data exploration.

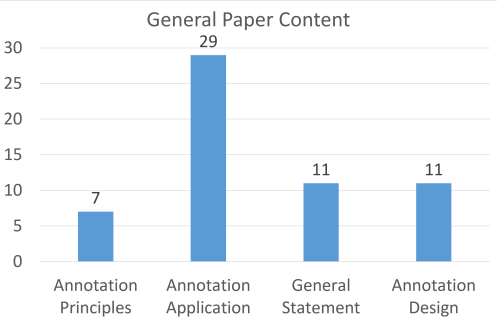

Figure 2.4: Accumulated results of the topics addressed.

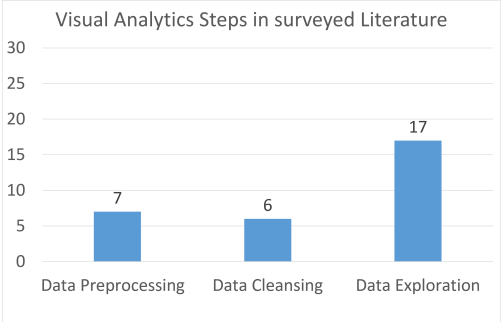

Figure 2.5: Accumulated results of the steps addressed.

Second, we have examined the characteristics of the used annotations themselves, which resulted in an overview shown in Figure 2.6. It can be seen that the annotations often follow a certain classification, which we have already used as a basis in the description of the literature in this chapter. Within this classification, certain characteristics of the annotations show up repeatedly, and can be assigned to the classification. With respect to the literature, Figure 2.6 shows this classification. The summary (Figure 2.7) reveals that the commonly used notion of annotations (user-supplied comments into a visualization system as well as classification of data) appears most often in the literature sample. As seen in the graphic, these comments are categories and free text, contain user information and outcome information, as well as result from alphanumeric input or automatic computation. While this shows a frequent use of annotations, our literature review equally shows that the use of annotations is much broader and more diverse. Other characteristics, such as graphical elements, provenance information, data information, markup, as well as selection and brushing are also used. Furthermore, the research shows that visual separation, layered visualization as well as visual encoding are used relatively evenly with a slight increase in visual separation. In this respect this classification seems to be reasonable for the examined literature. Nonetheless, all of these characteristics carry pros and cons. For the question "What are annotations?", *categories*, for example, consist of structured content and are not as suitable as *free text* for recording the thoughts of domain experts. Predefined category lists, on the other hand, can be easily processed by automatic analysis systems, e.g., for classification. Similarly, *graphical items* are well applicable to mark an area of interest. In the same way, different key characteristics for "Why annotate?" may be suitable in different situations. For example, annotations containing outcome information are handy during the data exploration, while annotations with data information may be helpful during data preprocessing. On "How to communicate annotations?" the concurrency situation between annotation and data visualization in combination with the relation between annotations and the respective data and/or visualization area has to be solved. While *visual separation* clearly distinguishes between the

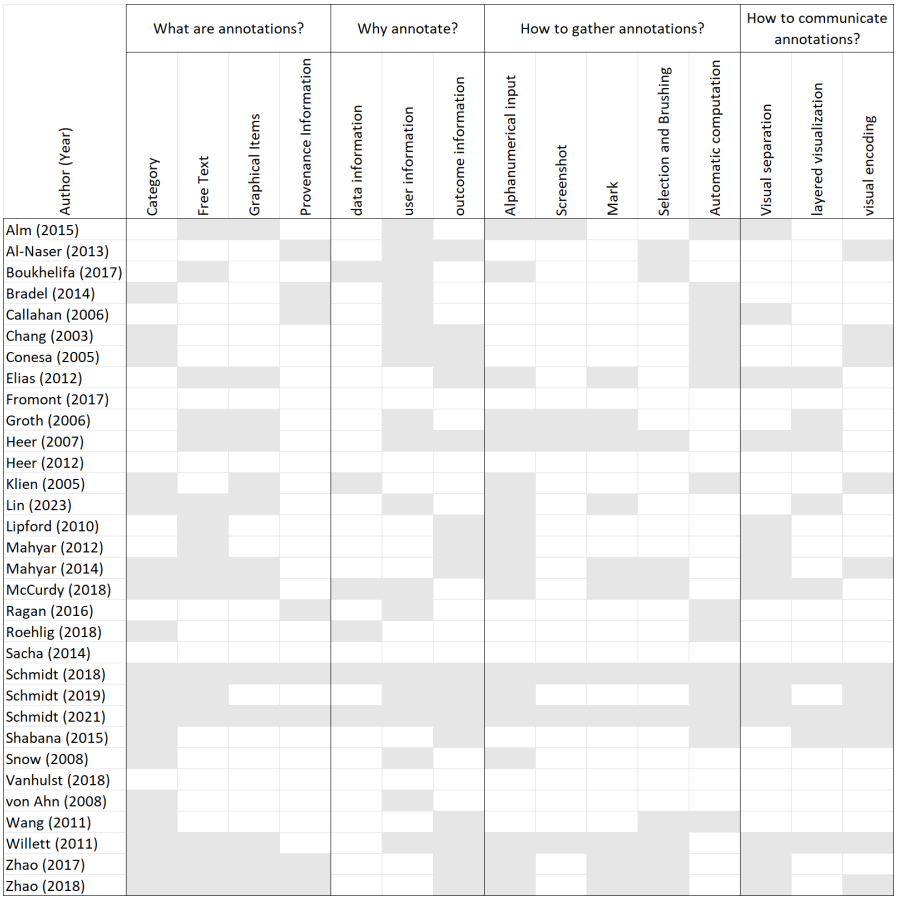

Figure 2.6: An overview on the existent characteristics of annotations in the example literature. The table shows which work supports which annotation characteristic (grey = supported). It represents the base for our discussion on the key annotation characteristics.

data visualization and annotation visualization, there is the need to find a way to relate the annotation to the data/visualization area it affects.

Visual encoding, on the other hand, allows a local visual connection of annotations and data within the visualization, creating a consistence of impression for users. Yet, the direct integration also implies the need to carefully adjust the visualization techniques to avoid an unwanted interpretation of annotation and data. This especially applies to unverified annotation content. These examples, to name but a few, elucidate the tensions and interdependencies between the annotation characteristics. We see that they have their own advantages and disadvantages, so that not all of them apply at all times. For this reason, it can be useful to reduce the key annotation characteristics, when applying annotations for a particular use case.

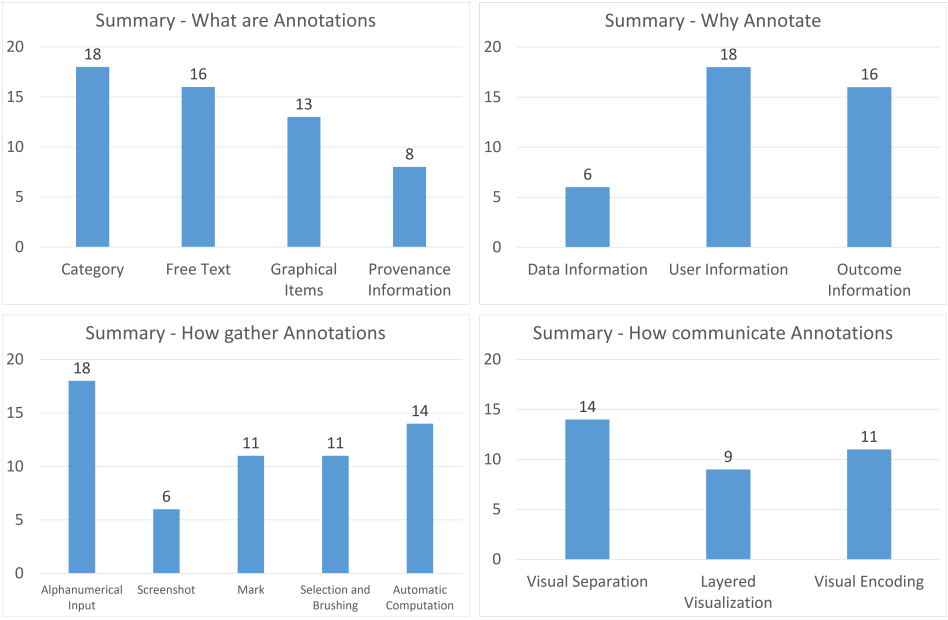

Figure 2.7: Summary of the annotation characteristics usage in literature.

However, it can be seen that annotations appear in literature in all the steps of visual analytics we address in this work. It is also recognizable that the focus has been primarily on data exploration; data preprocessing and data cleansing have hardly been addressed. The introduction of a model for systematic annotation generation could have a positive effect here.

Overall, we have found that there are a variety of approaches for the use of annotations with the existent characteristics, but a particular naming and classification of the characteristics is missing. Furthermore, the introduction of a model for systematic annotation creation, by reducing the key characteristics to suitable ones, could have a positive impact on the broader use of annotations in visual analytics.

In the context of this dissertation, we will close this gap by introducing a morphological box with an overview of key annotation characteristics as well as developing a model for the systematic use of annotations with a subset of the key characteristics, which we refer to as *suitable characteristics*.

3 Means of Annotation Characterization

Following the collection of existing annotation characteristics from literature in the previous section, we now introduce means of annotation characterization. First, we develop a morphological box to systemize the existing characteristics as key characteristics. We use the morphological approach as proposed by Zwicky [Zwi67]. He writes: "The morphological approach to discovery, invention, research, and construction has been conceived and developed for the purpose of dealing with all situations in life more reasonably and more effectively than hitherto. This is achieved through the study of all relevant interrelations among objects, phenomena, and concepts by means of methods which are based on the utmost detachment from prejudice and carefully refrain from all prevaluations."

In accordance with the structure in Section 2.2, we divide our morphological box into four rows. Each row represents one of the four basic questions on annotation characteristics. For each basic question we list the key characteristics and provide them as an overview in the morphological box as shown in Figure 3.1. With the help of the box, it is possible to (i) get an overview on key annotation characteristics for each basic question when annotating and (ii) allow a free combination of these characteristics. This allows annotation developers to characterize annotations for many use cases in visual analytics.

What are annotations?	category	free text	graphical item	provenance information
Why do we annotate?	add data information	add user information	add outcome inormation	
How to gather annotations?	alphanumerical input	screenshot	marks	selection and brushing	automatic computation
How to communicate annotations?	visual separation	layered visualization	visual encoding	

Figure 3.1: The Morphological Box with the key characteristics of annotations derived from the example literature. The table shows the key annotation characteristics after their classification into the four basic questions on annotations.

The second part of this chapter introduces a novel annotation model. It enables the identification of suitable annotation characteristics out of the full number of possibilities. It does so by respecting the requirements of a possible use case and visual analytics. In this step-by-step process, the reduction of the key annotation characteristics to suitable ones for a specific use case is performed first. Subsequently, matching annotation design aspects are identified, building the prerequisites for the design and implementation, either into an existing visualization system or a new holistic approach.

On the basis of the model, the final part of this chapter describes the application of the model to show its usefulness. This is done on our novel workflow to combine different steps in visual analytics into a user-in-the-loop workflow.

3.1 Key Annotation Characteristics

On the basis of the literature analysis in Section 2.2 as well as our own research on annotation characteristics ([SRS18; Sch+19; Sch+21]) we are able to systemize the existent annotation characteristics by a morphological box as shown in Figure 3.1. For a better understanding of the theoretical concepts in this work, we briefly depict each annotation characteristic.

What are annotations?

As outlined in Section 2.2, we can distinct between the four characteristics *category*, *free text*, *graphical item*, and *provenance information*.

Categories have the purpose to intake structured, class wise annotations by allowing the choice of one or more values from a list. The list does not have to be closed. With the generation of *free text* users can integrate nearly any text-based information. We understand free text as a set of alphanumerical items, possibly, but not necessarily put together as words, numbers, sentences and texts. In contrast to free texts, *graphical items* allow the integration and presentation of glyphs, pictures, and similar objects for the purpose of annotation. The last characteristic of annotation in our scheme is *provenance information*. It represents information concerning the visual analytics process, typically recorded in the background and can consist of interaction history as well as presentation history or the preservation of different states of the analysis.

Why do we annotate?

Typically, there is a reason why additional information should be added during visual analytics. Based on our studies, we see the need to distinct between three major reasons: (i) annotate information about data, (ii) annotate information from

the user, (iii) annotate information about the outcome. These reasons will be explained in detail below.

Adding information to the data is for the purpose of explaining data or providing information on data operation circumstances. The second reason to annotate, *adding user information* is the integration of the users' knowledge. This typically concerns domain knowledge, considered necessary or helpful to be added to the system. Our third reason to annotate is *adding outcome information*. It is information that has emerged as a result of the analysis and can be externalized with the help of annotations.

How to gather annotations?

The annotation gathering process describes the acquisition of annotation information by the system or the input of the user. This concerns mainly the logical and technical aspects of recording annotations in a visual analytics system. Respectively, the gathering process (annotation generation) can be either automatic or manual. In this way, the key characteristics from "What are annotations?," and "Why do we annotate?" imply different prerequisites for the gathering techniques. Even though some gathering techniques are often associated with specific characteristics from "What are annotations?," such as alphanumerical input for free text, most of the gathering techniques can be used for more than one characteristic. Selection and brushing, for example, can apply to the selection of category annotation as well as to drag and drop a graphical item for annotation.

In this respect, there are five key gathering characteristics from Section 2.2: (i) *system computation*, *alphanumerical input*, *graphical item*, *screenshot*, and *selection and brushing*.

The first characteristic is *system computation*, which represents a gathering method to automatically retrieve information during a visual analysis by computational means. That information can be stored with some annotation overhead data for later use (e.g., externalization).

Alphanumerical input means the collection of texts or numbers. Texts can be free text manually recorded by the user or automatically generated words, added text from further sources, text documents, and others. That is, the information recorded, is encoded in the alphanumerical input, in the form of numbers, words or texts.

Graphical items on the other hand, are glyphs or other forms of graphics added to the visualization system. When added to the visualization system, they often mark a point or area of interest in the visualization. The recording of graphical items include the location in the visualization and/or the data reference as well as describing information, such as shape, size, etc..

Taking a *screenshot* is a common method, to generate an image of the screen at a certain point in time, e.g., during the analysis. That screenshot can be either from a single view of the visualization system or of the screen as a whole, including, e.g., a parameter settings panel.

Selection and brushing are common interaction techniques. For annotation they can be used to intake additional information from the user, e.g., allow an annotation selection from a predefined list. Brushing is the selection of one or more elements from the visualization on the current screen. In this respect, the selection itself can be the annotation process, or the selection can be an initial step to record further data from the user.

How to communicate annotations?

The communication of annotation is important, yet difficult. The goal of communicating is to provide the gathered annotation information to the user. Here the question arises, whether or not the communication is intended or even needed to be done together with the data in the same visualization or even in the same view. Secondly, if the communication is intended to be digital, through displays (in contrast to, e.g., printing, or offline recording), it necessarily uses display space, requiring appropriate means for managing the display space. In this respect, we have found three key characteristics of annotations for communication:

Visual separation means that the annotation content is displayed distinct from the original visual analytics views. This means that the annotation communication is either done in a separate view within the data visualization system or even, after externalization, within its own visualization system, particularly dedicated to annotations.

Visual encoding directly integrates the annotations into a visualization view. Here the same encoding types (e.g., colors, emphasis, etc.) apply both to the underlaying data and the annotation, so careful distribution of available encoding techniques must be ensured to allow distinction between data and annotation. Yet, visual encoding also allows to assimilate verified annotations into data visualization. Here, the annotation information is shown with the same encoding as the original data, so that there is no intended difference. This can be used to, e.g., hide incorrect data and/or show corrected data together with the original data, and thus, avoid misleading visualization due to errors in the original data. This method has to be used with caution, as it does not differentiate between original data and annotations, so that the risk of showing erroneous annotations must be minimized through appropriate verification methods.

Layered visualization means that the location of the annotations is within a data visualization view, yet the encoding techniques differ, so that it is visually possible to distinguish between annotation and data. One example is the possibility to

enable/disable the visibility of annotations, to allow for switching between showing and not showing them.

3.2 The Annotation Characterization Model

In the previous section we depicted the key annotation characteristics organized in a morphological box. Within this box, various combinations of annotation characteristics are possible on a concrete annotation design. Yet, as shown in Section 2.4, not all of these combinations are useful for all cases. Open questions are, for example, "Which annotation characteristics are useful for a particular use case?," "Which annotation characteristics are suitable for a specific data visualization?," and "How are they designed?." We consider these open questions on annotations as an *annotation problem*.

To solve this problem, we develop a novel model to allow for specific annotation characterization and design for a particular use case. We introduce and describe the model in this section.

3.2.1 General Model Design

Our model has sources which provide us with the necessary information for solving the annotation problem. The three sources are (i) the key characteristics from the morphological box, (ii) the requirements from the use case, as well as (iii) the requirements from either an existing visual analytics system or a holistic system to be developed. Moreover, our model consists of a process with two steps, which work with the information provided to solve the annotation problem. The two process steps are (i) reducing the morphological box and (ii) designing the annotations. Figure 3.2 demonstrates the visual interpretation of the sources and steps.

3.2.2 Assessment of the Requirements

While the morphological box with key annotation characteristics is available from the start, the use case requirements and visual analytics requirements have to be identified individually for every use case. Inspired by the Nested Model from Munzner [Mun09], first and foremost, the use case is analyzed and discussed in close cooperation with the domain experts, in order to identify the needs and use case requirements for the specific annotation problem. Here, the motivation for annotation (Why annotate?) and annotation type (What are annotations?) from the use case perspective is clarified. For example, if users want to integrate classification information, category can be a suitable characteristic. The use case requirements also contain the requirements from data, as different annotation characteristics may be suitable. Annotating quantitative data, for example, could be beneficial with

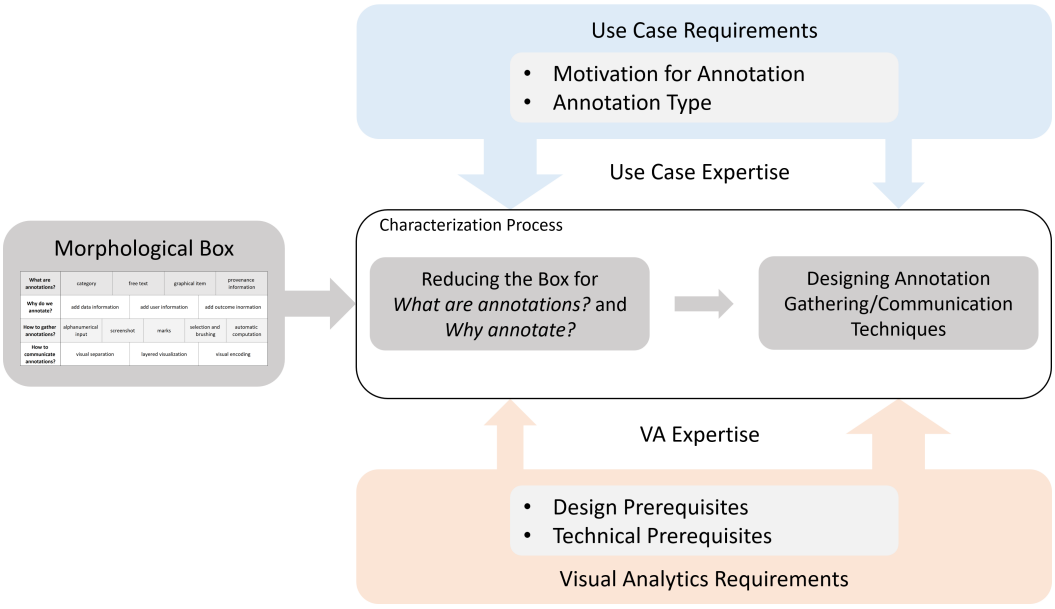

Figure 3.2: The novel annotation model to solve an annotation problem. First the applicable annotation characteristics are identified. This is done in respect to the use case, visualization parameters and key characteristics from the morphological box. Second, the annotations are designed in respect to the use case requirements and the (possibly existing) visualization system.

category annotations, while the annotation of qualitative data could be useful with free text annotations.

Additionally, the visual analytics requirements need to be brought to light ("How gather?," "How communicate?"). To derive the visual analytics requirements depends on two scenarios: (i) there is an existent visual analytics system, or (ii) a new visual analytics system with annotations is created. For option one it is important that there are existent restrictions due to existing visualization techniques, where annotations must fit in. If no system exists, then these questions can be answered in a holistic approach with the system design and the design aspects can be considered in combination with the data visualization needs. While the visual analytics requirements for annotation communication are similar to the ones for visual analytics, the input process for the annotation information (the gathering) has its own requirements. Usually, these requirements do not apply for visual analytics systems, which solely support presentation and interaction functions. The main questions are, how and where to input the additional information as well as how and where to store the additional information. They are individually answered for each use case.

3.2.3 Process Application

At this point, we have the morphological box and the requirements from both the use case and visual analytics as an input for the process application. Taking that as a base, the four basic questions from the box can be answered in a two-fold way, following the characterization process in Figure 3.2. In the first step of the process, it is important to reduce the key characteristics from the morphological box to suitable characteristics for the first two basic questions "What are annotations?" and "Why annotate?." As the use case defines the information type and content, the use case requirements are more dominant in this step of the process, while the visual analytics domain plays an assisting role.

For example, the annotations are *categories*, if the domain experts want to classify their data and *free text* if the domain experts want to input their own explanations. In a similar way, it is up to the domain experts to decide, if they want to integrate *user information*, e.g., explanations from the user on data peculiarities, or outcome information, e.g., a postulate on an identified feature in the data visualization. Still, advice on the requirements engineering process from visual analytics experts is helpful. For example, the visual analytics specialist can explain the particular differences between, e.g., *user information* and *data information*. This is why we integrated the support of visual analytics experts into the first step of the process as seen in Figure 3.2. In this way, visual analytics experts with knowledge on the annotation characteristics model can help domain experts to articulate and classify their needs into the annotation characteristics.

On the second step in the process, the design of annotations, the visual analytics requirements dominate. Still, the prerequisites from use case (e.g., user needs) are taken into account. During the design process, the data visualization requirements and annotation visualization requirements need to be merged and concluded into a combined visualization concept. This concerns the gathering and communication process for the annotations, which are often interdependent. One aspect, for example, is, if a visualization system already exists or a new holistic system is developed. For example, visual encoding may be difficult, if there is an existing visualization system, so that layered visualization or even visual separation may be the only options. Other visual analytic aspects are, for example, the number and type of views or visualization techniques needed. Here, a suitable gathering technique needs to be chosen. So, it can be beneficial for users to manually mark the border between clusters within a scatter-plot visualization, while it may be more efficient to automatically compute maxima and minima for bar chart. Nevertheless, it is also important to consider the needs of the domain experts, as they need to accept and understand the visual analytics system.

When these design decisions under all the requirements have been made, we have suitable annotation characteristics with fitting design aspects for an annotation problem at hand. However, it is necessary to consider the implementation of the

design as well. This implies that annotations are provided to users with respect to their analysis workflow. This will be examined in the following section, where we demonstrate the application of the annotation characteristics model on the user-in-the-loop workflow for visual analytics.

3.3 Annotations in the User-in-the-Loop Workflow

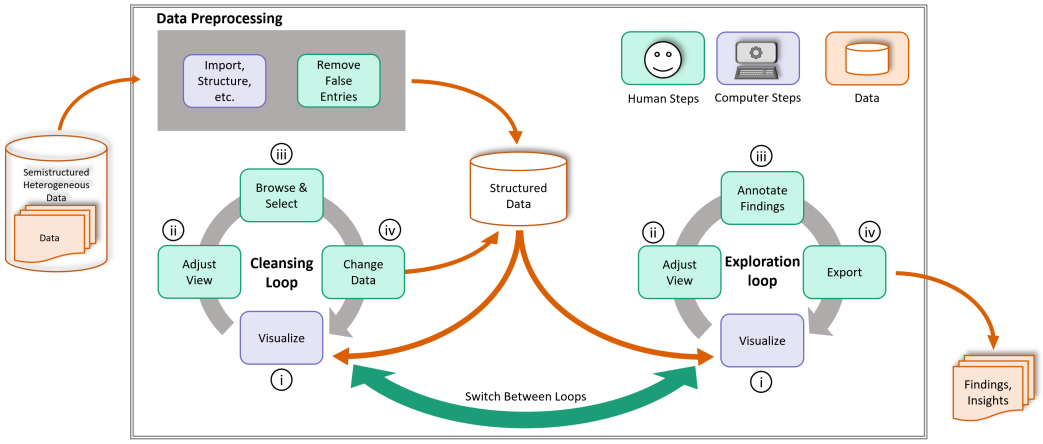

Figure 3.3: Our user-in-the-loop workflow to prepare, cleanse, and explore data from the medical domain.

In our work *Combining Visual Cleansing and Exploration for Clinical Data* [Sch+19] we introduce a user-in-the-loop workflow with visual analytics methods, which can be beneficial for the task fulfillment in the domain of ophthalmology in clinical environments. During the workflow development and application on a use case in the medical domain, we integrated a variety of annotations. Deriving the annotation characteristics with the help of the model from Section 3.2 revealed specific annotation characteristics for each step, which will be depicted in this section.

To provide an overview on the workflow as shown in Figure 3.3, we provide a brief description. The workflow is divided into three fundamental steps: (i) data preprocessing, (ii) data cleansing loop, and (iii) data exploration loop. The *data preprocessing* ensures that the raw data are correctly imported, structured, merged, and presented to the user in an appropriate way. The *data cleansing loop* supports the identification, removal, or amendment of erroneous or missing data points. In this step, the content of the data points is examined. Within the *data exploration loop* the experts can examine interconnections between data dimensions and identify, select, annotate, and/or export their findings. While the workflow foresees an initial run of the data cleansing loop, it deliberately allows a switching between the cleansing and exploration loop at a later stage. This gives the possibility to iteratively amend and/or improve the data at any point of the analysis.

In the following, we show for each step in the workflow how our annotation characterization model reveals suitable annotation characteristics for each individual step in that workflow. These suitable characteristics can be used as a base when the annotation characterization model is applied on a use case, which we do in Chapter 4.

3.3.1 Suitable Characteristics for Data Preprocessing

The data preprocessing step has the goal of collecting and structuring necessary information from all available data sources for a particular use case [Sac+14]. Figure 3.4 illustrates this process. The data may stem from different sources (e.g., data

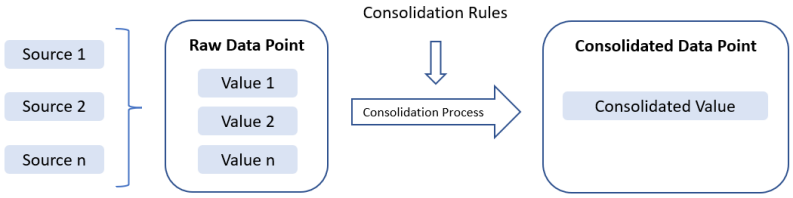

Figure 3.4: The process for data consolidation during data preprocessing. The data from all sources is collected into a raw data set, where data points can be redundant, which means, they have more than one value. These values are consolidated to one data point value for each data point by applying user-defined consolidation rules.

bases, electronic devices, manual recording, etc.). When the data from these sources are merged to raw data points, redundancies and discrepancies within the data may appear, as there may be several data point values from the different sources for a single data point. For a structured data analysis, these redundant data point values are often consolidated to one value. To solve consolidation conflicts in case of discrepancies between the data point values, rules have to be applied, with which the final consolidated value is retrieved. With the application of the consolidation rules, it is possible to generate a consolidated and structured data set.

To derive suitable annotation characteristics, we apply our annotation characteristics model. This means that we first derive the requirements from the use case and from visual analytics perspective. As users want to understand the results of the consolidation process, the first requirement is to record the circumstances of the consolidation process. This includes the recording of the available data sources, the applied rule, the result of the consolidation, etc.. As the overall data are often of large quantity, the second requirement is to automate the preprocessing process as much as possible and keep the manual input low.

To derive the visual analytics requirements, which mainly concern the gathering and communication characteristics, we look at the technical and visual aspects of data preprocessing annotations. The gathering characteristics requirements are driven

by the need to record transparency information, which means to record information, which is created within and during the data consolidation process. These are, for example, the different redundant values and the consolidation rule, which is applied to derive the consolidated data value. To record this, it is required to integrate the annotation gathering directly into the consolidation process.

The communication of annotations is driven by the preprocessing need to understand the consolidated data. We can provide this by making the circumstances of the consolidation, such as the consolidation result and rule, transparent. To allow users to associate the transparency information to the respective consolidated data point, one requirement is to show it in the same location within the same view as the consolidated data. Yet, this must be done carefully, as the consolidated data itself needs to remain in the focus for assessment and quality control by the domain experts. This means that, the annotation visualization is done only if necessary and beneficial for understanding the data consolidation process.

Applying these requirements on the morphological box (see Figure 3.1 in Chapter 3), we can derive suitable annotation characteristics. For the requirement to record consolidation information, we assess the type of information to be collected. In the case of data preprocessing, this concerns free text (e.g., number of sources or merged data point values) and category information (e.g., discrepancy information "yes/no," or source names).

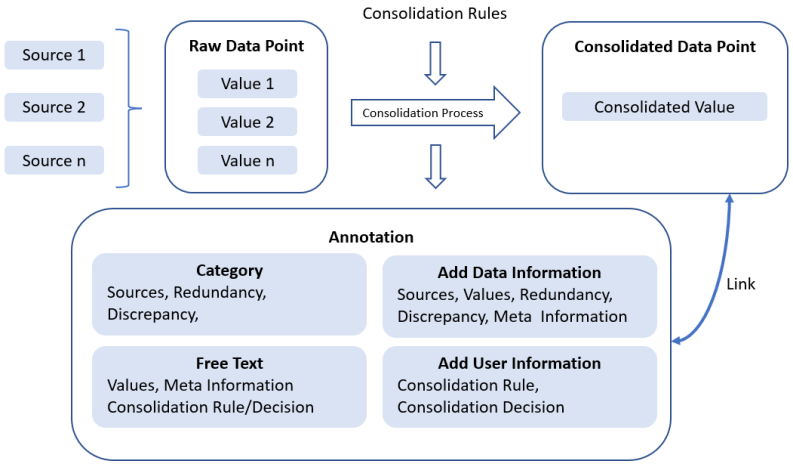

Figure 3.5: Annotation generation during data preprocessing. The annotation content is generated during the consolidation process. All necessary information to fulfill the use case and visual analytics requirements is generated and saved in a structured way by applying the annotation characteristics.

To define the characteristic for the second basic question, we need to know the reason for annotations during data preprocessing. As one use case requirement is the documentation of the data consolidation actions, we know that the annotations

record information on the data itself (redundancy, source information, etc.). This means that the reason to annotate is to add data information. Additionally, for the predefined rules, which originate from the users' knowledge, adding user information applies.

For gathering (third basic question) automatic computation comes into view, as the annotation generation should be as automated as possible at this point. To allow for a later reference, the annotations are recorded in an annotation structure and mutually linked to the respective data point. In sum, we can integrate the annotation recording process into the data consolidation process as shown in Figure 3.5.

Finally, the decision on annotation encoding is driven by the requirement to associate the results of the applied rules visually, locally where the consolidated data is visualized. This is the case for visual encoding. Explaining information, on the other hand, could lead to too much clutter or occlusion, potentially disturbing the experts' view on the consolidated data. For that reason, visual separation, e.g., on demand, is a suitable choice.

3.3.2 Suitable Characteristics for the Data Cleansing Loop

Data cleansing usually has the goal to reduce the number of missing, misleading, or wrong data points; short - to "correct dirty data" [Gsc+14]. In the data cleansing loop from the process model in Figure 3.3 this is achieved by (i) browsing through the data, (ii) selecting and changing the data, (iii) update the visualization, and (iv) further adjustment on the view as the basis for the new cleansing-loop run. Within this loop, the experts' knowledge in combination with context information is required, such as nearby data points, so that manual corrections are necessary. With the help of the visualization of the consolidated data, experts identify the data points to be cleansed. On these data points they apply their editing operation, resulting in a cleansed data point. The cleansing process is illustrated in Figure 3.6.

Figure 3.6: The consolidated data points are the base for the data cleansing loop performed by experts. During the cleansing process, users edit the data by deleting or changing erroneous data points and/or adding missing data points. In doing so, a cleansed data set is achieved.

If fully allowed and undocumented, these corrections can completely alter the original data, and thus bear the risk of introducing new errors and leaving the user unaware of changes made. To reduce these risks, annotations can provide information on when, how, and by whom, which data values have been edited, so they can be used for documentation.

In applying our model, we first extract the use case and visual analytics requirements for the data cleansing annotations. The need for annotations to document the changes made, results in the use case requirement to record and display the aforementioned information associated to the amended data point. If this recording is done manually, it is time consuming [Jin+17]. Therefore, the second use case requirement for data cleansing annotations is to keep the effort for annotation information recording low.

From the visual analytics perspective, the documentation of the data change should be made available on demand, under the prerequisite that the documentation is locally linked to the respective change. This is to be able to provide the necessary information to the user where needed.

With these requirements, we now can apply the process in our annotation characteristics model and assign suitable annotation characteristics to the data cleansing annotations. As the source names and user comments are potentially unknown, they are free text, while the altering information (add, change, delete) is a closed list of possibilities, and thus, category information. In the same way, the source and altering information add information on the data itself, while the user comment on the change typically contains additional information from the user, who is responsible for the data cleansing by using his or hers existing domain knowledge. This answers the basic questions on "What are annotations?" and "Why do we annotate?."

Finally, we need to identify the suitable characteristic for the questions "How gather annotations?" and "How to communicate annotations?." For gathering the annotations, the requirement to keep the effort as low as possible comes into view. We can support this, if we can record the necessary information "on the side" during the editing process as McCurdy et al. [MGM18] have shown. Therefore, we integrate the annotation gathering into the data cleansing process as shown in Figure 3.7.

The visual analytics requirement for annotation communication during data cleansing is, to allow for identifying and understanding the changes made in the editing process. A good way for identification is the local highlighting of the change, which refers to the characteristic of layered visualization. In contrast to visual encoding, which means a similar encoding as the original data, layered visualization represents an additional layer on the visualization. Within this extra layer, highlighting visualization techniques are possible, which significantly differ from the original data. A further advantage of layered visualization during data cleansing is the feature to hide the extra layer, and thus, hide the annotations to allow for a view on the

cleansed data only. This has the advantage that the annotations can be viewed locally connected to the data, if necessary, but hidden at any time, if a judgement of the cleansed data itself is necessary.

The additional information on the source and potential user comments, on the other hand, are often only necessary on demand and therefore more suitable for visual separation. This may be an extra view, shown on demand, or a separate area of the screen.

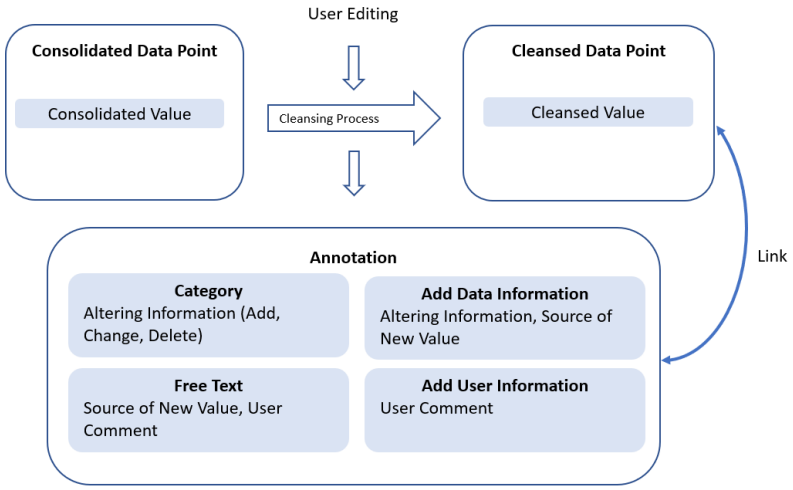

Figure 3.7: The consolidated data points are the base for the data cleansing loop performed by experts (adding, changing, deleting). During that loop, data cleansing information is recorded and stored within an annotation. That annotation is linked to the cleansed data point.

3.3.3 Suitable Characteristics for the Data Exploration Loop

Sacha et al. [Sac+14] see the goal in data exploration to identify findings and gain insights. They state that "a finding is an interesting observation made by an analyst using the visual analytics system. The finding leads to further interaction with the system or to new insights." In our user-in-the-loop workflow (Figure 3.3), we support the exploration in loopwhise execution of (i) visualizing, (ii) manual view adjustment, (iii) annotating, and (iv) exporting. This can be done in several iterations, also in a switchback to the data cleansing loop, to allow for data quality and exploration improvement.

Annotations at this stage have been used to support this process by locating the findings, documenting the findings, and externalize the findings and, if applicable, the gained insights. From the use case perspective, one requirement is the free exploration and documentation of possible findings. The other requirement is, to

make the findings and possible insights permanently available, even beyond the end of the exploration session. This is represented by the externalization.

The visual analytics requirements during data exploration are threefold. First, the use of annotations for identification means that the annotation on the screen must be suitable for marking these findings. Second, the description of the finding by experts as well as the discussion between experts needs to be supported by suitable means of user input and presentation of these inputs for later reference or collaborative environments. Third, the integration of marks and comments should be with low cluttering and occlusion to avoid the disturbance of the data exploration.

With these requirements, the annotation process for data exploration includes the annotation generation into the data exploration loop as shown in Figure 3.8. This ensures that all necessary information can be recorded.

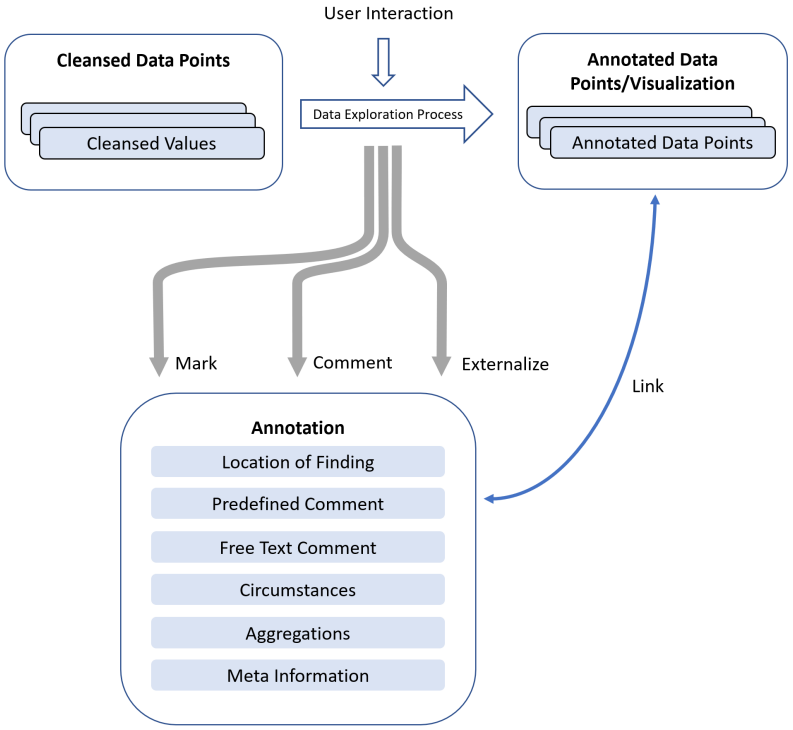

Figure 3.8: The concept on annotation gathering during the data exploration loop. Via dedicated interaction methods annotations are recorded. That concerns either marking, commenting, or externalization.

Applying the annotation characterization model, the premise for the first basic question is the requirement to support all kinds of user input in a free exploration environment. To fulfill this requirement, all key characteristics for "What are annotations?" (category, free text, graphical item, provenance information) are also suitable. Examining the second basic question, the reason to annotate will be in

most cases to add outcome information. This especially applies to aggregations (e.g., maxima & minima of the data) and predefined comments, e.g., labels for classification of images. Nevertheless, we have seen in the literature (e.g., Willet et al. [Wil+11]) that user information is gathered as well, especially in discussions between users within a comment section.

The suitable gathering and communication characteristics are driven by the requirement to allow the localization of findings on the one hand and the discussion between users on the other hand. For the gathering, the requirement to preserve and externalize the findings and insights plays a major role. Thus, manual interaction incidents are injected into the data exploration itself, allowing users to specifically annotate their current needs via different gathering or externalization options. In that way, we assign alphanumerical input to free text comments, screenshots to circumstances, which we consider the current state of the exploration and view within the visual analytics system. Marks are used for findings, while users select the predefined comments and aggregations are automatically calculated.

To minimally disturb the exploration process, but clearly mark findings, the marks are displayed as layered visualization at the location where they were recorded. This preserves the located findings and draws attention to them, when necessary. The comments, on the other hand, are displayed in a separate view to avoid cluttering. To still enable the linkage between the visualization and comments, screenshots are used.

3.4 Discussion

The result of Chapter 3 is threefold. First, we are now able to classify existent annotation characteristics from literature into a morphological box. This box provides a sophisticated toolbox with key annotation characteristics. Yet, we also understand that our list may not be complete, as there are a plethora of use cases, from which not all could be considered in this work. As the base of the morphological box are existing characteristics from literature, lacking the purpose of being a closed set of complementary characteristics, a clear distinction of characteristics sometimes is difficult. As an example, it is difficult to decide, if the annotation of a changed data value during data cleansing is a data information or user information, if the user inputs that information. Therefore, it is always important to consider the context of the use case, when deciding on the annotation characteristics.

To address this issue, we introduce the second contribution from Chapter 3, the annotation characterization model that supports the development of annotations for a specific use case with the help of the morphological box. With its purpose, the collection of the requirements both from the domain as well as visual analytics, it provides the necessary information for reducing the morphological box to solve the specific annotation problem. Still, as the variety of annotation problems is

practically unlimited, a guiding example could be helpful, to conveniently apply the model.

This is the goal of the third contribution. Here, with the user-in-the-loop workflow, we enable an interlinked application of data preprocessing, data cleansing, and data exploration.

With the help of the morphological box and the annotation characterization model we are able to establish a sophisticated set of suitable annotations with specific characteristics for each step in our workflow. An overview of the derived set of annotation characteristics is shown in Figure 3.9. Even though we are aware that the user-in-the-loop workflow was designed for a use case in the medical domain, our derived general annotation characteristics may well serve as a base for other visual analytics problems, where the data preprocessing, data cleansing, and data exploration steps are applied.

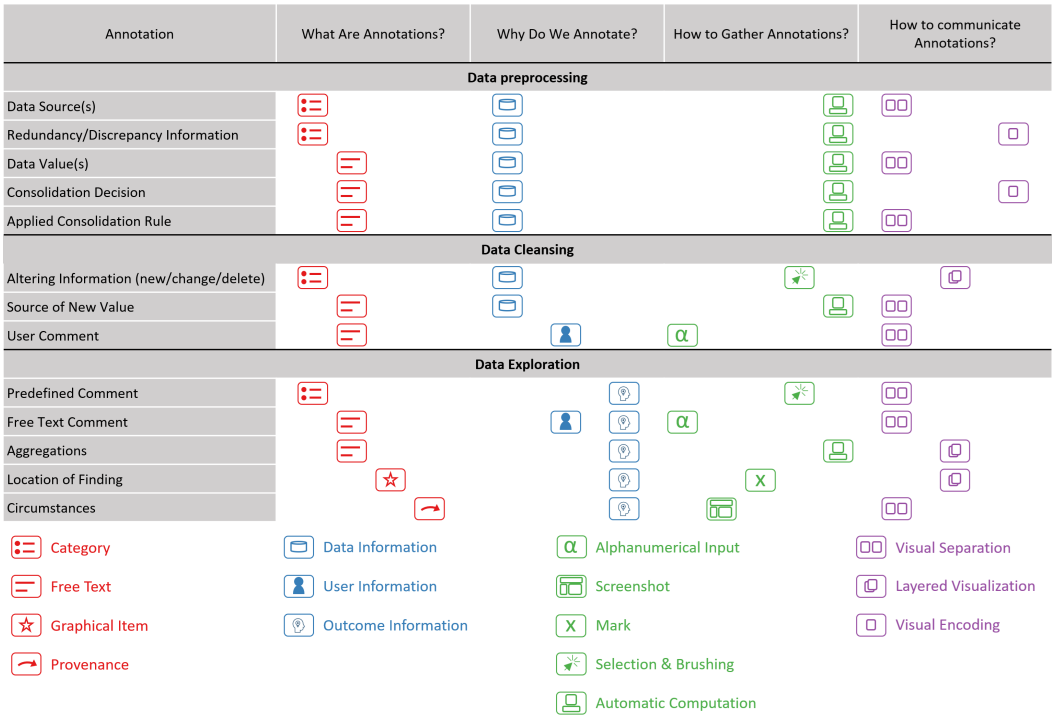

Figure 3.9: Suitable annotation characteristics in the steps of our user-in-the-loop workflow.

We know that the user-in-loop workflow does not include other steps in the analysis, such as verification loop or knowledge generation loop. Furthermore, there are other methods, such as Card et al. [CMS99] and Keim et al. [Kei+08; Kei+10] to describe the means of visual analytics.

While the theoretical concepts from Chapter 3 allow to solve an annotation problem in visual analytics, there is no general solution for a design development of these annotations. So far, we have a solution for what suitable annotation characteristics apply, but we still need to find how a fitting design can look like. Developing that design on the basis of heterogeneous medical data will be done in the following chapter.

4 Annotation Design

Our model, defined in the previous chapter, describes the dependency of annotation design on requirements from the use case and the visualization. Respecting this, our goal for this chapter is to apply the model in developing a more specific annotation design solution for our particular use case. This use case is the analysis of heterogeneous, redundant, and erroneous medical data as described in Section 2.1. To tackle this issue, we design annotations for different steps in the analysis. This means, we develop a particular solution in distributing the visualization space and visualization techniques between the data and annotations. This process is designed (in accordance with our model) as a participatory process in close collaboration with our domain experts. With these prerequisites annotation design faces further challenges, such as the limited visualization and interaction space, as well as the visual linking between data visualization and annotations.

Respecting these challenges in combination with the model application, we create an annotation design and discuss the result with respect to the visual analytics steps assessed in this work ([SRS18; Sch+19; Sch+21]).

4.1 Collaboration Setup

For a better understanding of the derived design concepts, we briefly describe the collaboration setup with the domain experts. We cooperated with a total of three retinal experts by using observation on the job, interview, and thinking-aloud techniques. The first goal was to understand the working day of a physician, so we performed an observation on the job over three days in an eye care center, where the visualization experts observed the everyday working process of the retinal physicians. This was particularly useful in combination with the thinking aloud technique, as the experts could explain the necessity of a domain related task and their motivation. Yet, especially with patients present, this was not always possible. For that reason, we conducted interviews after each domain related task, to clarify questions raised and reflect the task and all involved data acquiring systems like slit lamp examination form, etc.. In the aftermath of the observation, the visual analysis experts designed a first draft of the visual analysis tasks. It took six months and several additional interviews to determine a final version of these tasks. There are several reasons for that: (i) the visual analysis experts recorded the domain related tasks in high detail, including some unrelated and misleading information, (ii) the

providable data changed several times, (iii) both the domain experts and the visual analysis experts had to define a common ground of understanding. Especially the last one was challenging, as the domain experts favor straight forward approaches with sparse color and form usage, which has a major influence on our approach.

4.2 Data Preprocessing Annotation Design

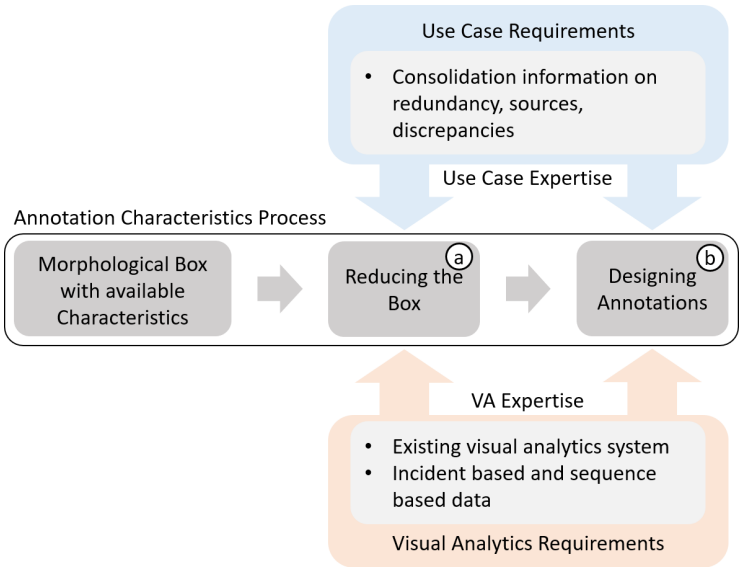

Figure 4.1: Recall of the model from Chapter 3 now applied to data preprocessing. First step reducing the morphological box (a). Second step designing the annotations (b). This is done with specific requirements from data preprocessing.

The first step for which we develop an annotation design is data preprocessing. As outlined, annotations in this step have the goal to inform the user about the circumstances of the consolidated data. This means that we make the user aware of existing redundancies, undertaken consolidations, and possible discrepancies. For this reason, we tailor the annotation characteristics in accordance with our model to these specific needs of the use case.

Our use case contains the data depicted in Chapter 2.1. The data stem from patients of an eye care center, gathered from different devices. They are heterogeneous, partially redundant, and potentially contradictory medical data.

Our prerequisites for the annotation design in this step are the general considerations made for data preprocessing annotations in Section 3.3. Here we find that annotations can be automatically recorded, including supplemental information on the consolidation process, and presented to the user with a reference to the original data.

Taking this as a base, we apply our annotation characteristics model. Figure 4.1 shows an overview on the model adapted for the data preprocessing step in the context of medical data. First, we will reduce the sum of possible annotation characteristics from the morphological box to useful ones under the premise of the given requirements from the use case. This represents step (a) in Figure 4.1. Step (b) covers the design process, influenced by the use case requirements and visual analytics principles.

4.2.1 Reducing the Morphological Box for Data Preprocessing on Heterogeneous Medical Data

We have shown which annotation characteristics from the morphological box are suitable for data preprocessing in general. Here, we discuss the reduction of the morphological box in order to specify the annotation characteristics for our particular use case.

In our use case we need to consolidate the data, meaning to gather information from different sources of the clinical management system, additional data tables and retrospectively digitalized documents. As a result, we have large amounts of heterogeneous data with redundant and possibly contradictory data values. Our experts have the ability to judge each of these data values in terms of its reliability when they know the source of the data value. On the other hand, they do not have the time to process all the data values manually. Thus, our experts required an automatic preprocessing. Under this premise, we reduce the available annotation characteristics from the morphological box accordingly (see Figure 4.2).

What are annotations?	Category	Free text	Graphical item	Provenance information	
Why do we annotate?	Add data information	Add user information	Add outcome information		
How gather annotations?	Alphanumerical input	Screenshot	Marks	Selection and brushing	Automatic computation
How communicate annotations?	Visual encoding	Layered visualization	Visual separation		

Figure 4.2: The morphological box with suitable annotation characteristics for the preprocessing step tailored to our use case. Suitable annotation characteristics are highlighted in yellow.

What are annotations?

In our medical-data consolidation process, we gather information about the source of the data point, redundancy information, and consistency information, meaning whether there is a discrepancy between redundant values. As we have large amounts

of data values and respective information, we see the necessity to make the information automatically processable. Here, predefined categories are useful to indicate different states of the data values. This applies to the data states *redundancy* ("yes"/"no") and *consistency* ("consistent," "contradictory," "no redundancy") in our use case. So, our first suitable annotation characteristic is *category*.

Free text, on the other hand, has the inherent property of allowing all possible characters, which makes it difficult to process automatically. As automatic processing is one of our requirements and we do not need further explanatory input in the data preprocessing step, we decide to not include free text annotations.

Graphical items during data preprocessing can be useful for highlighting missing or contradictory data in the visualization. This applies in our case, as we want to show the annotations directly with the data, to emphasize consistent as well as contradictory data, respectively.

Provenance information normally is gathered by recording user actions during the analysis. As the analysis itself has not started at this point, that characteristic is not applicable.

Why do we annotate?

Our model provides three different answers to the question "Why do we annotate?": "add data information," "add user information," and "add outcome information." In the sense of the morphological box, adding data information refers to information directly related to the data values, adding user information refers to an interactive input of the user, and outcome information is the result (finding, insight, conclusion, etc.) of a user analysis of the data.

During our data preprocessing step, we process each data point by automatically applying a set of rules to that data point. These rules will be described in detail in the next paragraph and were developed with the help of the domain and system knowledge of the ophthalmologists. Any result of these rules applied is related to the data values themselves. This can be, for example, a redundancy information, a consolidated data value, or source information. In our sense, this is *add data information*. As the automatic rules were developed on the base of the experts' knowledge, we can argue that we also annotate information from the user. However, we amend information on the data itself using automated rules from users, without manual interaction. As we understand the integration of expert knowledge as a manual interaction task, the application of automatic rules is not in the sense of adding user information in reference to our morphological box.

During data preprocessing data is prepared for analysis, while the analysis itself is not performed. Therefore, we do not have any outcome at this step and the purpose of adding outcome information does not apply.

How to gather annotations?

Gathering annotations during data preprocessing means to collect information on the consolidation process for each consolidated data point. In our case this implies thousands of patient related data points, for which the sources as well as the redundancy and discrepancy information has to be recorded. Doing this manually would require a tremendous amount of effort, unacceptable in our medical domain with experts who have limited time and are prone to frequent interruption by external requests. Consequently, the annotation gathering process needs to be performed automatically. Yet, certain information, such as prioritizing sources for solving discrepancies, need the domain knowledge of the user. For this reason, we create automatic rules prior to the data preprocessing step. These rules define and weigh existing sources, list valid ranges of values, cover specific exceptions, and provide grouping information to categorize the data. For example, in Figure 4.3 we

Case	argos for that date existent?	pdv for that date existent?	op for date existent?	argos_tl medication existent?	op medication existent?	medication op & argos_tl consistent?	will be automatically verified?	consolidated value
1	yes	yes	yes	yes	yes	yes	yes	argos_tl date & medication
2	yes	yes	yes	yes	yes	no	no	manual check
3	yes	yes	yes	yes	no	no	yes	argos_tl date & medication
4	yes	yes	yes	no	yes	no	yes	op date & medication
5	yes	no	yes	yes	yes	yes	yes	argos_tl date & medication
5	yes	no	yes	yes	yes	no	no	manual check
6	yes	yes	no	yes	no	no	yes	argos_tl date & medication
7	yes	no	no	yes	no	no	no	enable multiple selection
8	no	yes	yes	no	yes	no	yes	op date & medication
9	no	yes	yes	no	no	yes	no	discard
10	no	no	yes	no	yes	no	yes	op date & medication
11	no	no	yes	no	no	yes	no	discard

Figure 4.3: One of the decision matrices developed in cooperation with the domain experts. This matrix shows the medication to choose for the consolidated data value, depending on the existence of a particular source.

show one of three decision matrices developed in cooperation with domain experts. With the help of this matrix, the automatic consolidation process can decide, which consolidated value to choose for the medication for a specific date.

The matrix works well for a limited number of possibilities and sources. For cases with a larger set of possible sources, we develop decision making rules based on logic principles to consolidate redundant data point, possibly containing discrepancies. We consider a discrepancy within our data as differing data values for a single data point. To be more specific for our particular data, a data point can be the patient's vision (terminus technicus: visual acuity) for one eye at a specific point in time. It can be recorded in various ways by different personnel or devices. For example, it can be recorded automatically by the auto-refractometer, by research personnel during studies, and/or manually during the examination by a medical assistant. All values are stored within the clinical system. In our data set, these values would be assigned to the patient and marked with the different sources (manual measurement, auto-refractometer, etc.).

To consolidate these, we hereafter provide an example for the selection of a consolidated value v_c with a source list S_c for a redundant data point P_r.

This rule bases on a sorted reliability list S, defined in collaboration with the domain experts during the requirements engineering:

$$S = \{s_1, s_2, ..., s_m\}, \tag{4.1}$$

containing all available sources for a data point value. The reliability of the sources is decreasing with the index of the source:

$$s_i > s_j \ \forall \ i < j \leq m. \tag{4.2}$$

The reason is that some sources are more reliable than others (e.g., the automated visual acuity recording by the auto-refractometer is more reliable than the manually written value in the doctoral letter). With this reliability in mind, the domain experts decided for discrepancies, that they want to take the most often occurring value, or, if the number of occurrences is equal, the value from the most reliable source.

The data point P_r is a two-dimensional matrix with the dimensions values V and sources S_v: $P_r = \begin{pmatrix} v_1 & v_2 & ... & v_n \\ s_1 & s_2 & ... & s_n \end{pmatrix}$.

Each value $v_k \in P_r$ with $k \leq n$ has a source $s_k \in S_v$ with $S_v \subseteq S$. The fact that $S_v \subseteq S$ means that each element s_k has a correspondent element s_l with $s_k = s_l$. It is important to note that the elements s_k in S_v are unsorted in respect to their reliability, while their correspondent elements s_l in S are sorted, respectively. For this reason, the index l of the correspondent element $s_l \in S$ provides an information on the reliability rank of the elements s_k.

With these prerequisites at hand, we derive v_c as follows:

(1) If $v_1 = v_2 = v_n$, there is no discrepancy. The consolidated value

$$v_c = v_1 \tag{4.3}$$

and the consolidation source list

$$S_c = S_v. \tag{4.4}$$

(2) If there are two values ($n = 2$) and $v_a \neq v_b$ (discrepancy), we choose the value stemming from the source with the higher reliability in S:

$$v_c = v_a \forall s_a = s_x \in S; s_b = s_y \in S \mid \text{if } x < y \tag{4.5}$$

$$v_c = v_b \forall s_a = s_x \in S; s_b = s_y \in S \mid \text{if } x > y. \tag{4.6}$$

The consolidation source list contains the source of the consolidated value:

$$S_c = \{s_a\} \mid \text{if } v_c = v_a \tag{4.7}$$

$$S_c = \{s_b\} \mid \text{if } v_c = v_b. \tag{4.8}$$

(3) If there are more than two values ($n > 2$) and at least two different values $v_a \neq v_b$ in P, then we choose the value with the highest number of occurrences.

If the highest number of occurrences is 1, then we choose the value with the highest reliability. This means, we have $S_v \subset S$ with each element $s_k \in S_v$ having a correspondent element $s_l \in S$. Let $L = \{1, 2, ..., m\}$ be an index set with

$$S = \cup_{l \epsilon L} S_l. \tag{4.9}$$

In this respect the indices $\forall s_k \in S_v$ in S form an index set $L_v = \{l_1, l_2, ..., l_n\} \subseteq L$. Now, we can choose v_c:

$$v_c = v_k \text{ with } v_k = v_l \in S \text{ and } l = \min(L_v). \tag{4.10}$$

In combination with other rules, such as plausibility checks on the data value (e.g., visual acuity value cannot be below 0) the rules described above are executed.

Other examples of the automatic rule include ways to solve discrepancies via plausibility checks, such as checking the validity of a data point based on the interdependence and existence of different surrounding points and sources. Thus, certain surgery incidents (e.g., cataract surgery) are only plausible if there were certain preliminary investigations with corresponding results. So, the information about the surgery incident must not only be evident from the operation register, but also respective examination data from preparatory examinations must exist.

All of the rule content comes from the experts, who know their domain, the patients and the peculiarities of the clinical data management systems and working rules.

With the help of these rules, we are able to gather categorized annotation information on the consolidation process via *automatic computation*. These are, e.g., flags like "single source," "redundant," or "contradictory," the recording of the specific rule applied, the resulting decision, and the consolidation sources with each individual value.

This is in line with the general analysis of data preprocessing annotations performed in Section 3.3.

How to communicate annotations?

The communication characteristics from the morphological box describe the level of separation between the data and the annotations. We define three levels of separation: "Visual Encoding," "Layered Visualization," and "Visual separation." As our goal of communication during data preprocessing is to give direct feedback about the data consolidation to the user, we want to directly alter the data itself. For example, we want to emphasize contradictory data points to indicate possible problems in the data. This means to directly encode our annotation locally connected to the data points and altering the data encoding itself, as, for example, contradictory data values could lead to a misleading interpretation. For this type of annotation communication *visual encoding* is the associated characteristic. Yet, the annotations also contain detail information, which may not always be necessary to be encoded. Instead, it may be displayed on demand in a separate window, for which *visual separation* applies.

4.2.2 Designing Annotations for Data Preprocessing on Heterogeneous Medical Data

In the last sub-section, we derived specific annotation characteristics for our use case. This sub-section describes what the annotations with these characteristics will show and how this is done.

What annotation content do we show?

At this time, we define the content of our data preprocessing annotations. In accordance with the use case requirements, we want to record and communicate redundancy and consistency information. For the medical data in our clinical environment, we collect the specific redundancy and consistency content from the clinical information systems (e.g., device data, clinical management system, etc.) in collaboration with experts. With all the content listed, we design a structure that helps users to judge their data and contains the following information:

- Redundancy and consistency information:
  - Redundancy information informs about the number of sources. If there is more than one source, the data value is redundant.
  - Consistency information informs about the consistency of the data values. If there is no redundancy, it is "single source," otherwise it can be consistent (all redundant values for the data point are the same) or contradictory (varying values for the same data point).
  - Solved discrepancy information provides an indication, if and how a detected discrepancy could be resolved during the consolidation process by one of the domain-expert-defined automatic rules.
- Source information (e.g., clinical management system, study data)
- Supplemental information (e.g., author, date)

The reason to record the redundancy and consistency information is to see how reliable a data value is. As clinical data is prone to errors (see Section 2.1 for details), redundancy and discrepancy information add more transparency to the number of sources and the consistency of the values between these sources. This will aid experts in judging each data point concerning its reliability.

The source information provides the experts with information about the source of a data value. As there are rankings between the sources concerning reliability, the expert can receive more information about the reliability of the data value.

The supplemental annotation recording information gives experts a better understanding who initiated the consolidation process and when it occurred. This can help in case of questions and to better judge, if an update of the consolidation process may be necessary.

How to show the annotations?

At this point we need to decide how to show the annotations under premise of the previously undertaken steps in our annotation development. This means, we consider the existing visualization system, the use case requirements, the visual analytics requirements, the derived suitable annotation characteristics, as well as the information to show listed in the previous paragraph.

With the given characteristics, we now aim to develop a visual design that matches the medical data and the goals for data preprocessing. Following our model from Chapter 3, shown abbreviated in Figure 4.1 (b), we consider the requirements from the use case and visual analytics.

The visual analytics requirements are driven by the close link between the data and the annotations from the rule-based automatic changes made to the data.

The annotation visualization requirements concern the design of the annotations on the screen. This annotation design should amend the data visualization, so that the shown information on data is altered. The altering differs, depending on the annotation content, which is the consolidation information. This means that users are able to judge the consolidated data points, as their design depends on the consolidation information.

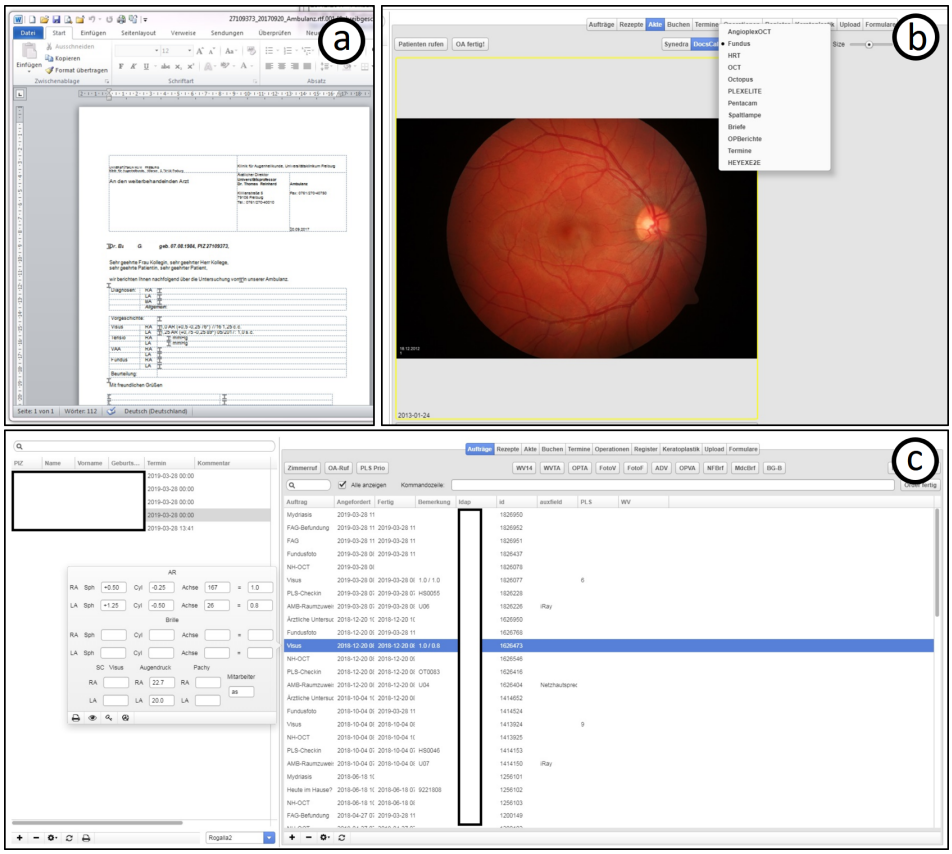

Figure 4.4: Three of the currently used tools by experts. Doctoral letters provide semi structured text-based information on an exam (a). Visual images of the patient's eye with exam information from medical apparatuses are examined with a proprietary tool software (b). The clinical management system provides search functions and table-based data presentation from the clinical data base (c).

In the current clinical management system, the clinical data are usually presented via tables as shown in Figure 4.4 (c). For each patient, there is an individual table with rows based on appointments, which means that the content time related data. This results in the request to base the consolidation on the table format but include all sources and patients.

From these requirements, we conclude that experts want to see the consolidation with the data point information in a structured and familiar form.

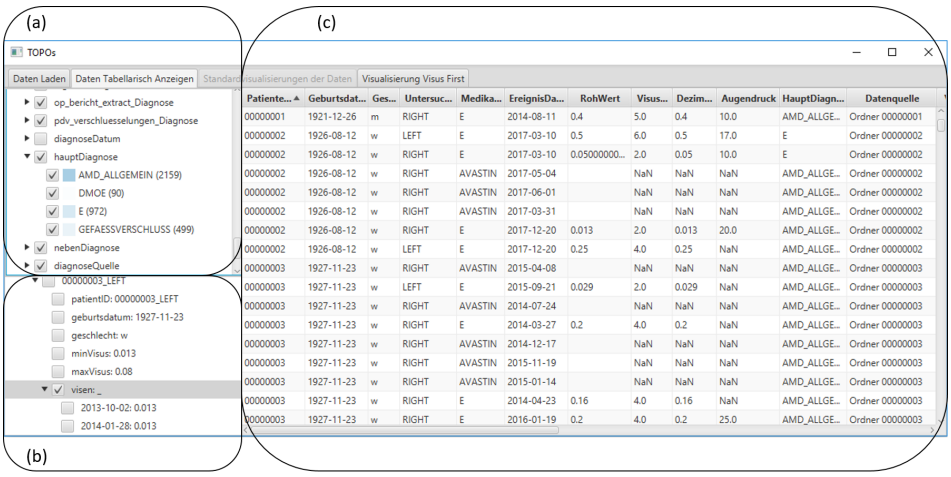

Figure 4.5: Our first design approach set the focus to the previous way of work by domain experts. Filtering options from the data are provided in (a) and (b), while the data and annotation information are presented in table form (c).

So, in our first design approach, we show the annotation information to the domain experts in table form as depicted in Figure 4.5. The table (c) is filled with the data points, amended with all available information from the consolidation. This includes redundancy, sources, and consistency, provided together with the respective original data point. This enables us to transparently show all information gathered during the consolidation process and present it to the experts in a familiar design. In doing so, we support the existing mental map of experts. Within the table the experts have their "identification elements," e.g., name/age/disease of a patient and/or date of exam, so that they can associate their memory on the patient triggered by the table information. With the amendment of our annotation information, the experts can integrate the annotation content within their mental map.

This design is simple, and text-based. It allows for understanding errors in the data, as discrepancies will lead to experts' explanations using their knowledge about the clinical system, the patient, technical devices, as well as circumstances of the data collection. For instance, the data collection procedure may vary due to different standards between everyday examination and study examination. So, if a discrepancy is detected, it can be easily resolved using expert knowledge. Yet, the detection of discrepancies itself is difficult, as experts need to scroll through many rows of a table.

For this reason, we integrated filtering functions (Figure 4.5 (a) & (b)). With the help of the filtering functions, experts can reduce the data shown in the table (c) in

two ways. First, the filter list in (a) shows all existent data points with their data values. For each value, its appearances are shown as an absolute number in the bracket and as a relative appearance represented by the opacity of the blue square-shaped glyph left to the data value. The higher the relative share of that value from all values is, the higher the opacity of the square becomes. This helps experts to spot outliers or main contributors, which then can be selected or deselected by the checkbox left to the blue square-shaped glyph.

Second, users can examine the data on patient level in the patient view (b). While the table (c) is incident based, which means that each row represents an incident (e.g., visual acuity measurement or anti-VEGF injection) for any patient, the patient view (b) is structured in a patient-oriented way, which means that the incidents from table (c) are sorted to the respected patient element. This means that, e.g., all visual acuity measurements for this patient are listed under the patient's visual acuity list, while all treatment incidents are listed in the treatment list for this patient. This allows experts to better examine an individual patient, which better reflects their patient based mental map.

With the filtering tool and patient view, experts can both reduce the table rows and explore the data on patient level to better detect interesting elements in respect to data consolidation. Nonetheless, the table design does not support visual linking of data point information and annotation information, and the amount of information is high, so that the data points shown on screen can only be few.

We therefore adapted the annotation design by integrating it into a data visualization approach. In collaboration between domain experts and visualization experts, a visual analytics system for the data was developed. Reflecting the requirements both from domain and visualization experts, we both kept the patient and eye-oriented visualization as well as the horizontal row-oriented visualization as introduced by Pleasant et al. [Pla+98].

An example of our visual design approach can be seen in Figure 4.6 (a). The visualization consists of several glyph-based elements. The main elements are vertically and horizontally oriented rectangles with different colors. The vertically oriented rectangles represent treatment incidents within the patient's observation period. These treatments are medication injections into the respective eye of a patient. The color indicates the specific medication and represents a color scheme for categories in accordance with the ColorBrewer tool from Harrower et al. [HB03]. The horizontal bar represents the current quality of the patient's vision. The higher the saturation of the bar is, the better is the ability to see of that respective eye. Using this visualization, we can reduce the space needed to communicate the patient data and integrate our annotation information on a visual basis.

With the reduced space needed and focus on the data itself in mind, we design an encoding, which shows the data consolidation operations only where necessary as illustrated in Figure 4.6 (b). Here, redundant and consistent values are not addi-

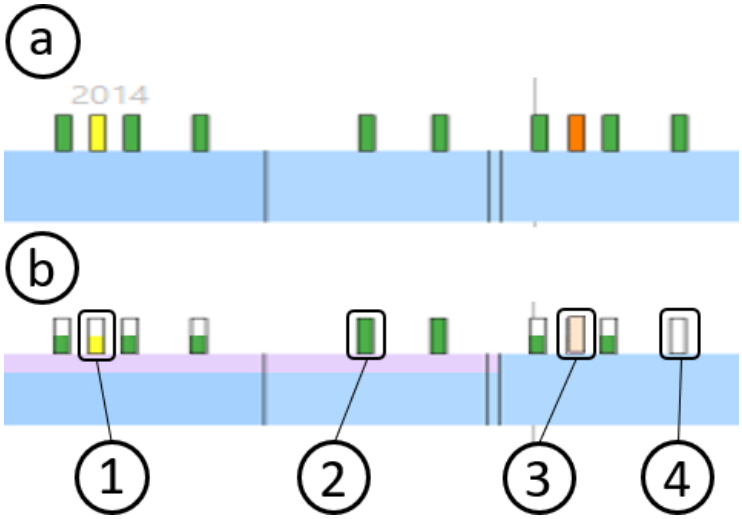

Figure 4.6: Our visual design of the incident and sequence based clinical data without annotation design (a) and with annotation design (b). Depending on the annotation content, the annotation design varies between partial color removal (1), no additional visual change (2), reduced opacity (3) or transparent encoding (4).

tionally marked at all (2). This is to reduce the visual load for the domain experts by reducing the indication for information not in the focus. Values, on the other hand, which are not confirmed by at least two sources are encoded as "half" (1) indicating that some confirmation is missing. The most severe change of the data encoding is given if a redundant data value contains discrepancies which could not be clarified by automatic rules (4). While the data point is still indicated by the glyph, the data value itself is not encoded at all, to avoid the presentation of misleading information. If an automatic rule could be applied to solve a discrepancy, we decided to show the value with less saturation. This decision, made in collaboration with domain experts, allows the experts to see the value in context with the patients' other data, while still hinting a lesser degree of confirmation.

This visualization allows for fast judgement of a large share of the consolidation results, but would risk cluttering, if we fully integrate the supplementary information, such as the sources for each data value. We therefore introduce the overview and detail principle to our visualization to show the additional information for a specific data point on demand, as shown in Figure 4.7.

One alternative would be the permanent integration of the separate view, e.g., on the right side of the screen. This would enable users to always see the supplementary data for a specific data point. Nonetheless, it would permanently consume screen space, while only showing that information for one or a few data points, while the users continue the analysis. Showing this information only on user interaction has

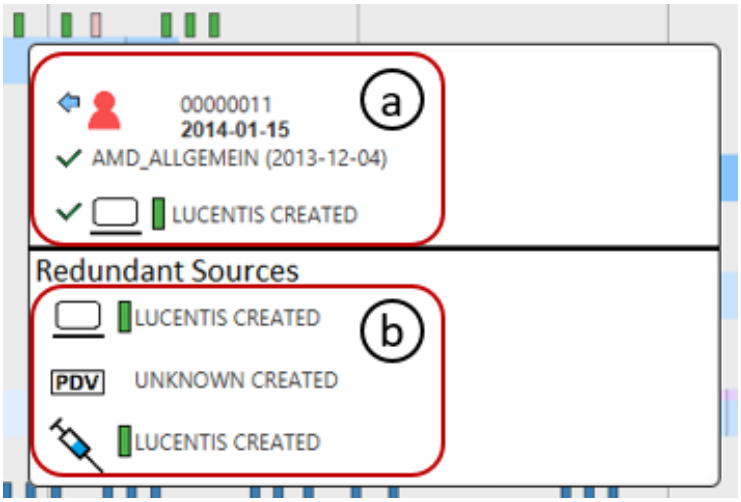

Figure 4.7: The annotation detail view. It shows key information about the patient and data point. This includes the side of the eye, sex, patient ID, date of data point, diagnosis, and consolidated data value (a). In the lower part, the detailed consolidation information is displayed (b). The sources and values for these sources are shown as well as their state within the data set.

the additional advantage that users actively choose to show this view, and thus mentally link the annotation information with the data point chosen.

The upper part of the detail view (a) allows users to recall the specific patient within the mental map of the expert. Talking to experts, they stated that they identify their patients from certain key information, such as name (which we replace by a generic patient id for data protection reasons), diagnosis, etc. We show this key information about the patient (such as sex, date of exam, and left or right eye) to allow experts to recall the peculiarities of this patient (remember patient and his/her history). Furthermore, the result of the consolidation information is displayed, to allow the integration of that information into the judgement of the experts.

Based on this information, all data available for a data point is displayed below (b). This gives experts the opportunity to examine and understand the consolidation result, as all sources (visually encoded), their respective data values (here: medication), and their state within the system (here: "created," can be "changed" or "deleted" at later stage). To ease the understanding of the consolidation, we redundantly show key data point information (a). In (b) all the supplementary information, such as source, value and data encoding information are provided.

After all information is retrieved, the user closes the additional view and returns to the previous overview on data preprocessing annotations.

4.2.3 Discussion

During the development of preprocessing annotations, we experienced the strong interconnection between the annotation characteristics and design and the domain experts' previous workflow as well as expectations for the VA system with annotations. As experts were used to look up single data sets within text-based data bases and documents, we designed our first data preprocessing approach accordingly. We already enhanced the existing workflow by bringing many data points together in one presentation with simple data visualization elements. We amended this system with text-based annotations. Even though this improved the data and annotation presentation and corresponded to the experts' expectations at this stage, the text-based annotations quickly revealed that a visual design is more suitable, so that domain expert expectations changed. We see one of the reasons in the familiarization of the domain experts with visual analysis during our project, and thus, their trust in the power of these. We consequently followed the expectations in changing the system and annotation design. This is reflected in our model, as we take the domain experts expertise and expectations into account, both during annotation characteristics and design development.

4.3 Data Cleansing Annotation Design

The second step in our analysis is data cleansing. The goal of data cleansing is the correction of the data. This includes the deletion, altering, and addition of data points or data values. Thus, during data cleansing, changes in the data are made. This step specifically applies to our domain, as the data are heterogeneous, redundant, contradictory and stem from various sources as discussed in Section 2.1. This leads to open data issues, even when the data preprocessing is successfully performed. In addition, the domain experts perform their daily tasks alongside their research work. This involves, for example, conducting consultations as well as exchange with colleagues on acute cases, resulting in frequent interruption of the research task.

Under these circumstances, we apply our annotation characteristics model from Chapter 3 and specify the requirements for data cleansing as shown in Figure 4.8. The derived requirements from domain experts for data cleansing are (i) to document the circumstances of data cleansing operations and (ii) to show (highlight) the cleansed data. This will allow domain experts to recognize and comprehend the changes made by other experts (collaborative environment) or themselves (interrupted environment).

Working with these requirements, we, hereafter, derive the specific annotation characteristics suitable for data cleansing in our use case and the respective annotation design for our heterogeneous medical data.

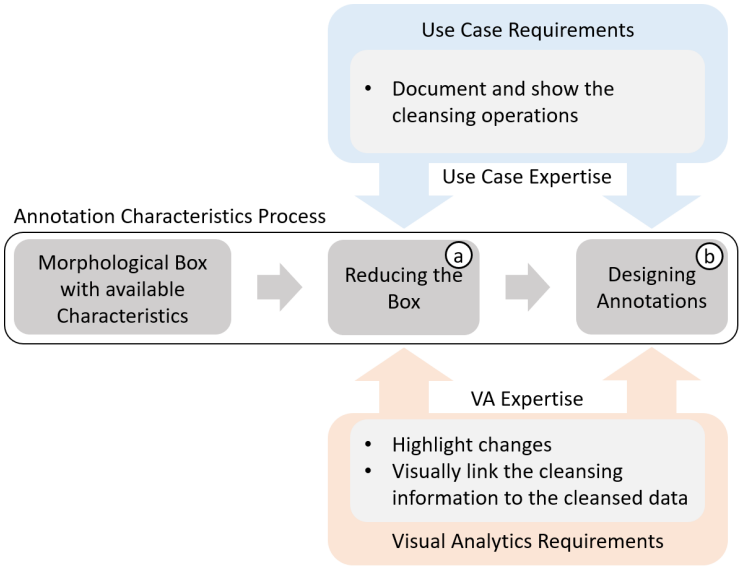

Figure 4.8: Recall of the model from Chapter 3 now applied to data cleansing. The first step is reducing the morphological box (a). The second step is designing the annotations (b). This is done with specific requirements from data cleansing.

4.3.1 Reducing the Morphological Box for Data Cleansing on Heterogeneous Medical Data

Driven by the identified requirements, we reduce the number of possible annotation characteristics for our data cleansing annotations. We do this by examining each of the questions from the morphological box individually, as already performed for data preprocessing. The result is described in the following and highlighted in Figure 4.9.

What are annotations?

As data cleansing often includes the cleansing categories "added," "changed," and "deleted," as it also does in our use case, we apply these categories for our annotation by using the *category* characteristics. This enables us to document cleansed data points and allows for differentiation between the changing operations.

In our case, the vast share of automatic data correction is performed during data preprocessing. For this reason, the data cleansing comprises mostly manual data changing operations performed by domain experts. To fulfill the documentation and explanation requirement, domain experts can explain their data cleansing operations in the cleansing step. As these explanations often include complex knowledge from the experts, we use the *free text* characteristic.

What are annotations?	Category	Free text	Graphical item	Provenance information	
Why do we annotate?	Add data information	Add user information	Add outcome information		
How gather annotations?	Alphanumerical input	Screenshot	Marks	Selection and brushing	Automatic computation
How communicate annotations?	Visual encoding	Layered visualization	Visual separation		

Figure 4.9: The morphological box with identified annotation characteristics (yellow-colored boxes) for the data cleansing step tailored to our use case.

As most cleansing actions are performed manually in our use case, we keep the additional effort to record the annotation information low. For this reason, we automatically collect explaining information, such as action taken, username, and time stamp, automatically on the side. We consider that as *provenance information*, as it gives an impression, what user action has been performed when and by whom.

Why do we annotate?

To identify the reason for annotation during data cleansing, we need to analyze the motivation of the experts during this step in the analysis. At this stage, the data preprocessing is finished, so that sorting, organizing as well as redundancy and discrepancy handling has been performed. So, the experts now want to remove and correct data content issues. That means, they view the discrepancies, which could not be solved, add missing values, or change wrong values within the data. For their annotation motivation that means, they want to integrate information on the data changes they perform (*add data information*), and the reason, why they made the changes, for documentation and better understanding in collaborative environments (*add user information*). As this is still the preparation of the actual data analysis, we expect no outcome information at this time.

How to gather annotations?

Gathering the annotation content during data cleansing is, recording in the category of the experts' cleansing operations, the provenance information from these actions, and the explanations from the experts concerning these operations. We gather the category and provenance information of the cleansing operations via *selection and brushing*. During the cleansing operation the experts need to select the correct operation (add, change, delete), which results in a respective category for the annotation. This is done for the provenance information in a similar form, as data point selections, user credentials, as well as the date and time are dependent

on the experts' actions within the system. The free text explanations from experts are recorded via *alphanumerical input*.

How to communicate annotations?

The communication of the annotations during data cleansing of heterogeneous medical data has two goals: (i) help the experts to see, if, and which changes have been made, and (ii) explain the changes by providing respective information. Nonetheless, the focus during data cleansing is on the cleansing task itself, focusing on showing the data to users. Showing the annotations here, is supposed to provide assistance without dominating the overall data view. So, we use *layered visualization*, to allow local connection of annotations to the cleansed data point, but also integrate the ability to hide the layer, for an undisturbed view on the data. If the explanatory information is needed for a specific cleansing operation performed, experts are provided with that additional information in a separate view using the *visual separation* technique. this will reduce the cluttering of the original data view and, similar to data preprocessing annotations, provide the necessary information when needed.

4.3.2 Designing the Annotations for Data Cleansing on Heterogeneous Medical Data

At this point, we have *category, free text*, and *provenance information* annotations which add *data* and *user information* and where derived via *alphanumerical input* and *selection and brushing*. Now, we describe the design process for these annotations.

What annotation content do we show?

In a first step, we define the annotation content for which the annotations with the respective characterizations are designed. For our analysis on heterogeneous medical data, we have the following use case requirements:

- Type of cleansing operation (add, change, delete)
- Indication on cleansing finished for all data on one patient's eye (yes, no)
- Documentation and explanation on cleansing operation (free text, supplementary information)

The type of cleansing operation is recorded during this operation and provides the visual analytics system with the information, what operation has been performed. In doing so, this information can be used later for indication. In the discussion with our domain experts, we learned that a remedy for the frequent interruptions in the cleansing process is a "cleansing finished" indicator on data groups (in our

case all data for a patient's eye). This helps users to continue their cleansing process at a later time without unnecessary effort to identify the point in the data, when the interruption began. Finally, the core content of the cleansing annotations is the explanation of the cleansing operation, so why the data point has been changed/added/deleted. This will help other experts to better understand the operation.

How to show the annotations?

Now we have all the information we need to develop the design of the annotations. Using this information, we identify the use-case-related and annotation-related requirements in accordance with the model from Chapter 3. This results in several aspects to be considered. On the one hand, the user has the goal to analyze the data in order to find data points to be cleansed. So, the focus during this step remains on detecting erroneous data-points and cleansing them, which the displayed annotations must not disturb. On the other hand, the annotations should provide sufficient information that is helpful in both judging the changes made and recognizing data points that have already been cleansed. Therefore, they must be visible and recognizable in some form. This leads to the design requirement, that both the

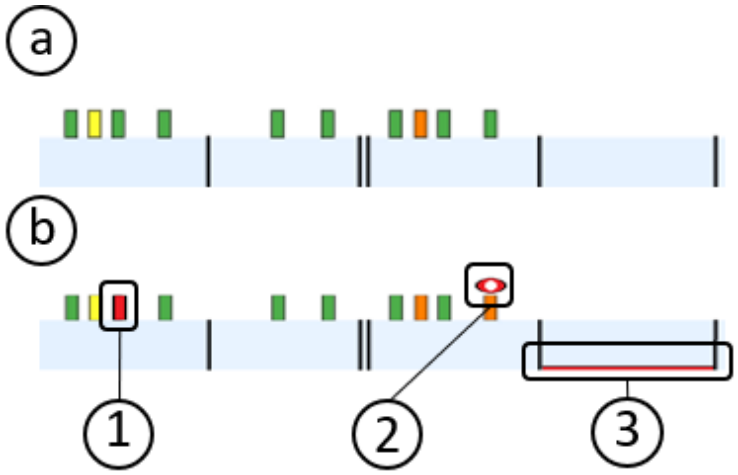

Figure 4.10: The cleansed data shown without visual encoding of annotations (a). As seen, there is no indication, and thus no recognition of the data changes made. In contrast, view (b) indicates the data changes via varying annotation encodings. The encodings can include deleted values (1), changed values (2), and added values (3).

original data and the annotations must be distinguishable on screen while shown at the same time.

To fulfill these needs, we use an overview and detail on demand technique, as shown in Figure 4.10 (b), similar to the preprocessing step. To avoid misleading

altering of the cleansing visualization, we provide an extra layer on top of the visualization with the highlighting information. While the data is encoded via rectangular glyphs with different colors, the annotations highlight specific parts with a distinct color and/or separate glyph. We show key information, such as the location and type of cleansed data points within the visualization area achieving the requirement to be clearly distinctive from the original data. In doing so, we are able to represent the meta information on the cleansed data without disturbing the original data representation, but still indicating locations, where data cleansing applies. This layer can be switched on and off by users, so they can decide whether the annotations are shown. This fulfills the requirement that the data visualization is fully available if needed, while important information for the user, such as location of cleansed data, is quickly available at first sight.

With this motivation at hand, we detail our design as shown in Figure 4.10 (1)-(3). We show the changes in the data performed during data cleansing. The original data visualization consists of glyphs with different shapes, representing time frames and data values for each eye of the patient. Within these data, particular data values can be changed during data cleansing. As per our requirements we highlight the location and type of change. The location is indicated via red colored glyphs and the type of change is communicated via the shape of the glyph. We decide to place the location locally connected to the changed data value as opposed to putting this information next to the visualization. This reduces the mental load for the user to translate any location information into the visualization coordinates, even though we increase the visual load within the visualization. We employ the same strategy for deleted data points, as we can use the annotation indication to cover the deleted data value and so remove the data value also from the user's view (1). The data values for altered data points, on the other hand, need to be seen and to be highlighted as changed. We therefore indicate changed data points with a circular glyph (2) next to the altered value. We use the gestalt law of proximity [BK53] by creating a nearby reference highlights the change and yet be able to show the data value. To avoid confusion with the original data, we use circular glyphs as a particularly distinct shape from the rectangular encodings of the data. For added values, we also use the highlighting technique to keep it simple. The main goal here is to simply highlight the location of cleansing actions. Yet, to distinguish between added and changed values, we indicate the location with an additional mark in linear shape on the encoded data value (3). In order to reduce the disadvantage of additional visual load due to the shown annotations, we support users in viewing the "pure" cleansed data, by including a feature to hide the extra layer with the cleansing annotation encoding (see difference between Figure 4.10 (a) - annotation layer hidden and Figure 4.10 (b) - annotation layer shown).

At this point, users are able to identify the cleansed data point, but have no further information, why this data point was cleansed. To fulfill the further requirement (the full understanding of the cleansing action), we provide annotation detail information, e.g., when and by whom which change has been made. The interest

of the user in our use case often concerns a particular data point which has been cleansed. Therefore, the user's focus normally lies within the data and annotation visualization, as he or she is interested in that particular point. For this reason, we decide to display the explaining annotation information directly at that local reference on demand (as a pop-up window). In this way, users can stay with their focus within the visualization even though the trade-off is the temporary occlusion of nearby data points.

As a specific need from experts for their highly interrupted cleansing process, we introduce the annotation information on already cleansed data. It is represented by an interaction field (checkbox), which is manually activated, when the respective data group (all data for one eye of a patient) is fully cleansed. Figure 4.11 (a). In

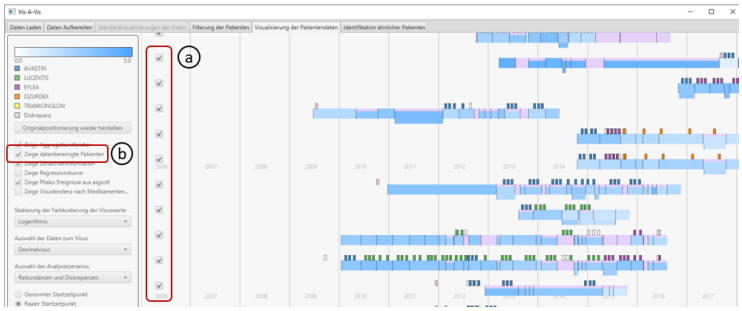

Figure 4.11: Design of the specific "cleansed checkboxes." By clicking on an individual checkbox for one eye of a patient, all data for this eye are indicated as cleansed (a). This allows for recognition of already cleansed data, even if no changes have been made. Via user-set preferences these data can be hidden to reduce visual load (b).

doing so, the user can annotate by simply clicking the checkbox, both indicating visually that this particular set has been cleansed and providing that information to the visualization system. This is especially helpful, as already cleansed data can then be hidden by the system on user choice Figure 4.11 (b), reducing the visual load on screen. This is also helpful in collaborative environments, as users can see which data have been cleansed by other users.

4.3.3 Discussion

Our annotation design for data cleansing alerts the user to several pieces of important information. The user knows at first sight where changes are made, and what type of change is made. By interacting with the visualization, the user can look into details, such as who made the change, and when the change was made. By integrating key information into the existing data visualization, we increase the visual load, which requires an additional effort for users to comprehend the visualization, yet, once the encoded information is understood, users can quickly

perceive the aforementioned information. We decided to choose an encoding, which increases the visual load, as our use case is located within a dense working environment, with only few experts, who need to be trained on the encoding. On the other hand, we use the overview and detail concept, to avoid too much cluttering. This highlights the use case specificity of the overall annotation design process. With the additional feature of feedback on the cleansed state within our design, we allow experts to conveniently complete the cleansing process, and thus, be able to fully and sufficiently prepare the data for the following exploration process.

4.4 Data Exploration Annotation Design

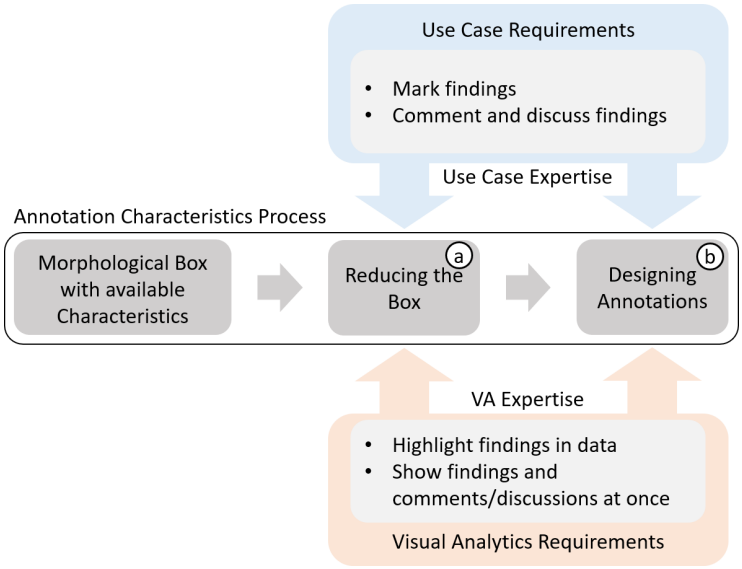

Figure 4.12: Recall of the model from Chapter 3 now applied to data exploration. The first step is reducing the morphological box (a). The second step is designing the annotations (b). This is done with specific requirements from data exploration.

After the data was preprocessed and cleansed, data exploration is the next step in our analysis, where the content of the data is analyzed. At this point, experts want to view the visualized data to identify findings and gain possible insights. Our annotations in this step have the goal of supporting the analysis by allowing to highlight identified findings, record user comments on these findings, which may contain insights, and provide the functionality of discussions between experts.

With these prerequisites in mind, we apply our annotation characterization model to reduce the morphological box to suitable characteristics and design the annotations with regard to the requirements from the domain as well as visual analytics. As

done in the previous sub-sections we use a customized model for data exploration as shown in Figure 4.12.

4.4.1 Reducing the Morphological Box for Data Exploration on Heterogeneous Medical Data

What are annotations?	Category	Free text	Graphical item	Provenance information	
Why do we annotate?	Add data information	Add user information	Add outcome information		
How gather annotations?	Alphanumerical input	Screenshot	Marks	Selection and brushing	Automatic computation
How communicate annotations?	Visual encoding	Layered visualization	Visual separation		

Figure 4.13: The morphological box with identified annotation characteristics (yellow-colored boxes) for the data exploration step tailored to our use case.

Reducing the morphological box for data exploration is a challenging task. Data exploration should be as free as possible, as the strength of this step is the combination of free data exploration with the vast knowledge of the experts. To combine both with wide annotation support results in a wide variety of annotation characteristics. For this reason, we concentrated on key goals of our experts. These are: (i) the identification and marking of findings within the boundaries of the existing visual analytics system, and (ii) the integration of comments and explanations for these findings, including the ability of collaborative discussion within the system. Beneath we discuss the corresponding characteristics combinations as seen in Figure 4.13.

What are annotations?

Our first objective is to allow annotations to highlight the position of findings within the data visualization. In the domain of ophthalmology, this means marking patient data, grouped for each eye, and evolving over time. In this visualization, we use *graphical items*, such as circles, which are used to tag the findings. These graphical items can be adjusted in size and location, to best fit the respective area and/or data. For the documentation, explanation, and discussion with annotations we use *free text*, as this allows users to express their thoughts and allow communication on the matter with others. Here, *provenance information*, e.g., screenshots of the visual analytics screen when the finding was identified, is of help, as it eases the explanation and supports the experts in recalling the circumstances of the finding.

Why do we annotate?

During data exploration we want to annotate the location of findings within the data visualization and explanations from and discussions between our domain experts, who are the users of the system. By highlighting the findings in the data visualization, we *add outcome information*, as the location of findings is one result of the data exploration step. Furthermore, we allow users to explain, comment, and discuss the findings. This integrates the knowledge from the experts, and thus, *user information* to the system. The data, on the other hand, is not amended at this time, as it has been preprocessed and cleansed in the previous steps.

How to gather annotations?

As stated before, we aim at designing the annotations as freely as possible during data exploration, in order to reduce the limits of data exploration. This means that we consider various ways to gather the annotations. To locate the findings in the visualization, we use *marks*, such as circles or ellipses. We show a circle example in Figure 4.14 (a). They aim at helping users to highlight the area of interest, often containing findings, within the visualization. As our domain experts work with a visual representation of their data, we identify and preserve the link of the markings to the underlying data currently shown on the screen. *Alphanumerical Input* is used to record user comments, explanations, and discussions. Users are free to input any text, numbers, or a combination, e.g., "This patient's visual acuity value has temporarily decreased from 0.75 to 0.25 after the cataract surgery. The decrease is not linked to the medication change." We use *screenshots* (Figure 4.14 (b) shows thumbnails of these screenshots) to ease the recall of specific analysis situations, reflecting the current data view and the state of the analysis. This provides a visual connection to an analysis situation and avoids the necessity for an alphanumerical description, thus, reducing the effort for experts.

Automatic computation may also apply for data exploration, for example for classification. Yet, we decided in cooperation with the domain experts that the number of possible criteria for classification is large and most of the time, the experts do not know in advance, which criteria they want to apply. We therefore rather implemented a manually operated filtering function and decided to not include automatic computation at this time. This allows user to filter, annotate, and then change the filter again in a loop-wise technique, which we proposed in [SRS18].

How to communicate annotations?

Similar to the other steps in our analysis, the communication of data exploration annotations is twofold. On the one hand, we want to highlight the location of findings in the data. In order to allow a direct visual reference between the highlighting annotation and the respective visualized data, we apply an extra layer on the data

visualization, so that we can communicate the annotations (graphical items) locally connected to the area of the finding. We use an extra layer, to avoid disturbances within the data visualization. We see that as necessary, because the visualized data is the base for domain experts for the analysis and to build their hypotheses. Any distortions may distract or interrupt this process. So, applying an extra layer allows an undisturbed view on the data by hiding the layer.

The other way to communicate our data exploration annotations is *visual separation*. As other work, e.g., Willett et al. [Wil+11] have shown, providing a separate comment space next to the visualization can conveniently support commenting and discussion functions. This is one of the requirements that our experts have. They need to explain and discuss their findings, either for themselves or collaborative with colleagues. It is important to simultaneously allow an exploration of the data, e.g., for further confirmation or exploration, and to view the comment area for reference to the current state of the discussion. This is necessary to maintain an interruption free thinking process during this phase of high cognitive load. From the visual analytics side it is important to distinct between the user and outcome information and the data.

At this point, we have original data on the one hand and assumptions and conclusions from users on the other side. Mixing these, e.g., by placing them in the same view with the same encoding, may bear the risk of data corruption, e.g., if an assumption is incomplete. While we integrate mechanisms for remedy, like commenting or discussing the assumption, we also decide to place them in separate areas on screen to clearly state their different sources.

4.4.2 Designing the Annotations for Data Exploration on Heterogeneous Medical Data

With the identified annotation characterizations, we are now able to integrate and display the annotation content for data exploration annotations in our use case. As already performed during data preprocessing and data cleansing, we define the content of the annotations we design in a first step. Finally, we use all the previous information on the annotation characteristics, the content and the requirements from domain and visual analytics experts to develop the design of the annotations within the visual analytics system.

What annotation content do we show?

In accordance with the use case requirements, the annotation content for data exploration annotations consists of

- the location of a finding within a displayed visualization
- a description of the finding

- an explanation/interpretation of the finding
- a discussion, containing content from one or more experts on the finding
- various supplemental information on the circumstances of the annotation, such as date, time, author

The location of a finding is identified by users through visually locating it within the visualization of a specific portion of the data currently displayed, e.g., the time series of patient data with a specific medication. A finding in this case could be, for example, a visual acuity improvement in combination with a particular treatment and an extraordinary incident. If this finding is marked, it will technically consist of links to all data currently shown on screen for reconstruction as well as the coordinates and geometry of the finding mark on the screen. If the user, for example, draws a rectangle on the visualization, the location and size of the rectangle is recorded, as well as all data links to the data points located within the rectangle. Data points, which are geometrically only partly covered by the drawn area are not included, as we interpret the partly coverage as the intention to not include this data point. The description of the finding contains user comments characterizing the finding itself, while the explanation/interpretation of the finding puts the finding into context of the analysis with possible conclusions towards the domain. The discussion content is comprised of explanations, comments, and/or judgements from other experts or the same expert at a later time. Finally, the supplementary information is the documentation of the annotation process, with date/time information as well as user information.

How to show the annotations?

At this point of applying our model, we provide the design details on data exploration annotations. Under the premise of the requirements from domain experts and visual analytics experts as given in Figure 4.12, we integrate the annotation design into the data exploration visualization. The first objective is to mark the findings in the visualization. Here, the annotation characteristics of graphical item and marks come into view. We use the graphical item characteristic to mark the locations of the findings as an extra layer in the visualization. The design of this mark is driven by the goal to allow users to quickly see the area of the finding within the visualization without distraction of the actual finding by the highlighting function. This means we need to draw the attention to this area without distorting the visual representation of the finding. Using the Gestalt laws “color” and “form,” we use circular glyphs with a color distinct from the data visualization as shown in Figure 4.14 (a). In order to reduce distraction, we use low thickness valve for the circle outline, so it appears less prominent and additionally allow hiding of the glyph layer. Again, the premise is “keep it simple.” This allows users to fully concentrate on the finding, after the glyph has fulfilled its task of locating the area.

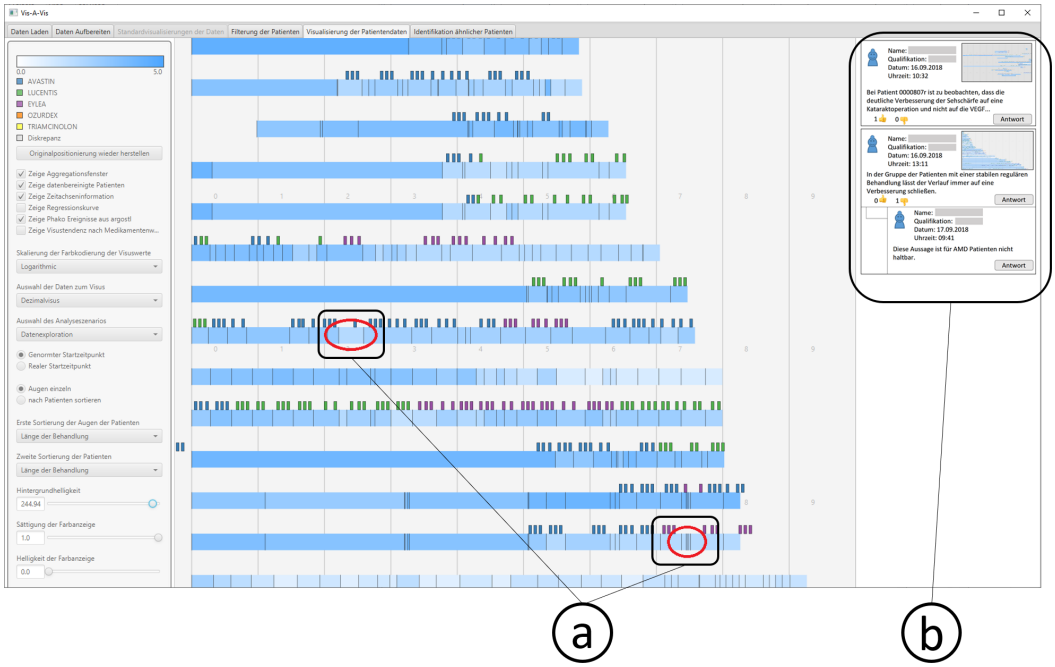

Figure 4.14: The data with annotation view during exploration. Annotations include markings in the visualization to highlight findings (a) and comments next to the visualization for recording of insights or discussion between experts (b).

The second part of data exploration annotation design is more complex. Here we integrate the free text and provenance annotations to add user and outcome information. This is to allow users to explain, interpret, and discuss their findings. As free text and screenshots require a lot of screen space, they compete with the primary visualization area and its functionality. As a conclusion from the reason above and the risks of mixing data and annotations during exploration, we avoid overlaying the visualization with the discussion annotations. A more suitable solution is a split screen, as Willett et al. [Wil+11] have shown.

We adopt this solution for our use case. As a result, we introduce a dedicated comment section on the screen, shown in Figure 4.14 (b). This dedicated area has the two functions to (i) present the data exploration annotation content and (ii) to provide interaction functionality for gathering the annotations. We design this section in a forum style, as this is well known to most users and offers the needed communication functionality, such as comments and discussions. The forum style has the goal to inform the user in a structured form on the comments/discussion annotations available. We therefore organize all entries in boxes (two in case of Figure 4.14 (b)), each providing the information on one annotation. Each entry box is divided into four sections.

The top section is the header. Here, we display the additional annotation data, such as date/time, in the left area of the header to aid users in mentally assigning the comment to an exploration situation and a user. The user is supported in recognizing the situation, in which the comment was made, with an additional screenshot thumbnail in the left part of the header, which is enlarged on user interaction. The free text section is located below the header showing the annotation's alphanumerical content, which usually is an explanation of the user's thought.

The interaction section is placed beneath the free text section. Here, users can interact with the comment by voting on positive or negative judgement or adding an answer to the comment. The fourth entry box section is only displayed if the user decides to add a comment. It is visually attached below the original comment as shown for the lower entry box in Figure 4.14 (b). In doing so, we reach a chronological readable structure, which allows users to follow the discussion in an ordered way.

As stated before, one disadvantage of visual separation is the missing local connection between the annotation and the annotated data section in the visualization. On remedy is the use of visual linking, as done by, e.g., Waldner et al. [Wal+10]. Using this feature would allow an instant recognition of linked annotations and affected area within the visualization. On the other hand, visual linking would also add additional cluttering and occlusion to the visualization, which we want to avoid. To overcome this issue, we add an interaction feature, that, if users need such a connection, highlights the area of annotation by increasing the thickness of the annotation circle in the data visualization on demand. This enables users to see which data area is connected to the current discussion. With this overall design, we reach a fully explorable data visualization supporting the implemented visual analytics features and additionally integrating annotations that allow marking, commenting, and discussing the results of the exploration.

4.5 Discussion

During the annotation design for our use case, we applied our annotation characteristics model to heterogeneous, redundant, and erroneous medical data. In doing so, we experienced that the structured development of annotation characteristics is beneficial for the annotation design. Based on the specific needs of the domain experts and visual analytics experts, the concurrency situation between the visualization of the original data and the annotations can be tackled. The same applies for visualization and annotation design techniques. Respecting this, we found that during data preprocessing, data cleansing, and data exploration on our data, a decent encoding of annotations is suitable.

Concerning our specific design process, we experience the challenge that the design possibilities for annotations are reduced if annotations are to be added to an existing

visualization system. As the visualization design techniques are fixed, annotations must adapt here, reducing number of techniques to be used, in order to reach discriminability, emphasis, or other design objectives. Therefore, it may be beneficial to choose an holistic approach, designing data and annotation visualization as a whole.

5 Visual Analytics Tool and Annotation Implementation

At this stage we have developed suitable annotation characteristics with a fitting design, tailored to our use case and visual analytics requirements. Now, we describe the integration of these suitable annotations into a visual analytics application. For all our implementation approaches, including both the analytics tool and the annotation extension, we used an implementation environment fitting to the use case demands. This means that we created a desktop application runnable on a standard personal computer system, commonly used in clinical environments. For the development we used the Apache NetBeans IDE Version 11 with the programming language Java and the JavaFX framework, which supports extended interaction and convenient GUI-object property dependencies. In addition, libraries, such as gson 2.8.2 for parsing JSON structures, and commons-math 3.6.1 for analytics operations, where used. The visual analytics tool is available in GitHub: `https://git.informatik.uni-rostock.de/cschmidt/topos-tool`.

For a better understanding of the implementation of annotation extensions in this work, it is important to elaborate the existing visual analytics tool. Therefore, the following section describes the general architecture of the visual analytics tool. Thereafter, the second section describes the implementation of annotations within the existing architecture, and the third section discusses the results of the implementation in the context of the annotation.

5.1 Visual Analytics Tool Design

Our visual analytics tool was published in Schmidt et al. [Sch+19] and has a general architecture as shown in Figure 5.1. It consists of the data layer (bottom left), analytics layer (bottom right), and user interface layer (top).

The *data management layer* reflects the necessity to parse, change, and store the data and annotations, which is an important part in our tool. It represents the interface between the internal data storage, the other layers and external data sources and thus controls all data exchange.

The main purpose of the *analytics layer* is to provide different aggregations of the data, sorting/grouping functions and parameter settings for the user interface. It

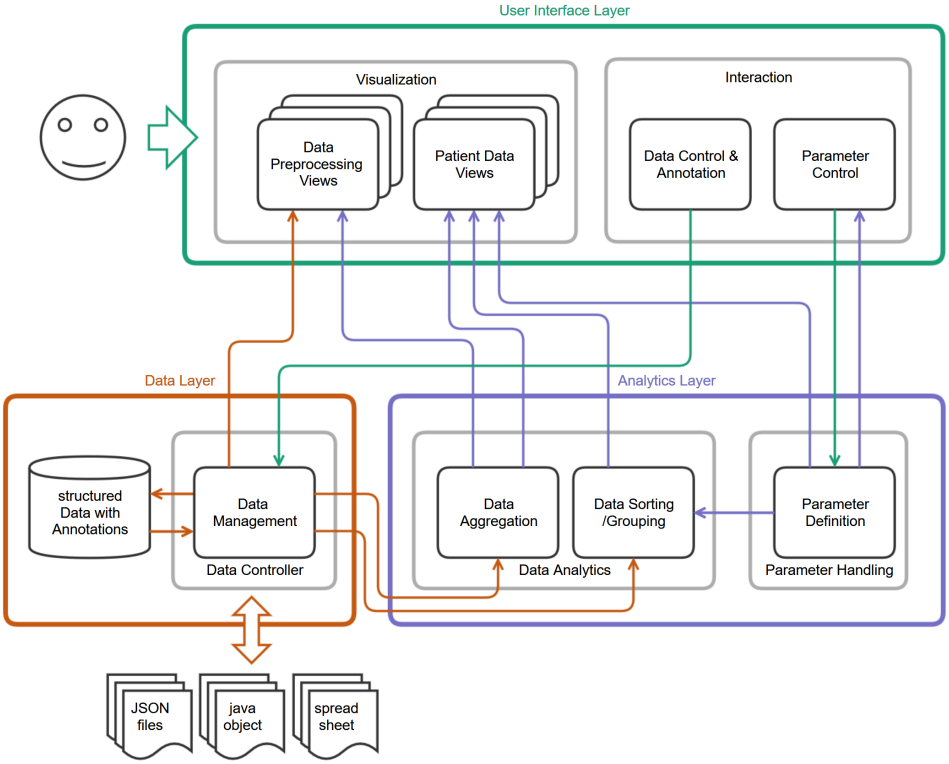

Figure 5.1: General architecture of our visual analytics tool with its three layers. The *user interface layer* (top) represents the interface with the user for data visualization and user interaction. The *analytics layer* (bottom right) holds functions for aggregation, sorting and parameter handling. The data layer organizes the import/export, parsing, and reading/writing of the data.

is designed as a separate component with interfaces to the other two layers. This makes it easy to implement additional or replace existing analytics functions.

With the given data and parameters, the *user interface layer* can present the data to the user in the visualization component. Additionally, the interaction component ensures that necessary user actions like parameter changing, data amendment, or annotation creation can be supported.

5.1.1 Data Layer

As data cleansing and data amendment are key elements in the workflow, we create a dedicated data layer for the tool. This layer is responsible for data preprocessing, the internal data storage, as well as the data amendments in the data cleansing and exploration loops. The *import and export of the data* is performed by a data parser, which uses predefined rules and mapping tables to convert the raw data

from the files into the internal structure. The mapping tables have been filled by the domain experts, so that user knowledge is integrated in our tool, refining some of the data dimensions already during the data parsing. For the export, the internal refined data structure is used to either create spreadsheet or a java object, both to be stored in the file system. They can be used for an import in the next session to continue the work with the data from the last session.

The *internal data storage* is done in two ways. One is the incident-oriented structure, for which the incident date in combination with the patient and the laterality for eye identification are unique and thus represent the key. For these incidents, all information gathered on this day are stored, which can be progression and/or treatment information such as visual acuity values, injection medication, cataract operation to name but a few. Due to the various data sources, this can lead to redundant information for some data dimensions. For these, additional dimensions are created to record the sources of the data value on the one hand, and store the information, if there is a discrepancy for the value between the sources. These additional data dimensions can be used during the data cleaning loop for emphasis. The gathered incidents are stored as a list of time-oriented multivariate data points, allowing a suitable response to show details on demand for a specific incident for a patient. The second structure is patient-oriented and holds the generated aggregations from the analytics layer, such as average visual acuity, number of injections, regression information and so on. In addition to the aggregated information, the patient structure holds lists of incident information for that patient, such as visual acuity measurement lists, with the dates and values that allow to quickly provide time series data, e.g., for regression calculation and visualization.

If the initial parsing is finished and the data is internally stored, all further *data amendment* is only done on user request. The amendment is always done for both internal structures. For the incident structure an additional dimension is used to mark the incident as "annotated." For the patient structure, a list, containing all amendments and a link to the affected data points for this patient, is stored. This list consists of annotation related information, such as the name of the user, the time-stamp, and the annotation content. This allows to permanently relate all user annotations to the respective data points. As soon as a data amendment request is carried out, the updated data is sent to all affected user interfaces.

5.1.2 Analytics Layer

The analytics layer supports the exploration loop and is designed in two parts. One is the parameter definition component, which controls the parameter settings through presetting functions, restriction definition, and automatic parameter changing. The other part of the analytics layer is the data analytics component, responsible for all data aggregation and sorting/grouping functions.

The *parameter definition* component supports rule-based specification on visualization parameters. With the presetting functions, parameters are set for the initial aggregation, sorting and presentation of the data. This supports users to focus on a specific analysis problem when starting the data analysis. They can be changed by the user and saved for the next session. The restriction definition ensures that the parameters are limited to valid values, e.g., only positive visual acuity values or valid time frames. The automatic parameter changing is a dynamic function, reacting to user actions during data exploration. So, for instance, if the data dimension is changed from a linear one to a logarithmic one, the color-coding parameter is instantly changed from linear to logarithmic, to ensure correct value encoding.

The *data analytics component* has various functions to aggregate and sort the data:

- **Appropriate scaling of data dimensions:**

  For certain numerical dimensions, the maximum and minimum values and/or the total number of occurrences (e.g., number of treatments) are derived, to determine scales for both, intra-patient, and inter-patient exploration. Furthermore, *conversion from logarithmic scale units to linear scale units* is performed. The visual acuity value in the data is given in the decimal scale (V_{dec}), as this is the common way to communicate between ophthalmologists. Yet, this scale is logarithmic, so the logMar visual acuity value (V_{logMar}) has been developed [BL76] to represent a linear scale. Additionally, we need the letter score (V_{letter}) which is used to measure visual acuity differences (see Wecker et al. [Wec+17]). We convert the decimal visual acuity values with the following equations:

  $$V_{logMar} = log_{10}(\frac{1}{V_{dec}}) \tag{5.1}$$

  and

  $$V_{letter} = -50 * V_{logMar} + 85 \tag{5.2}$$

- **Aggregation of patient data:**

  For intra-patient numerical data points, like visual acuity values and tensio values (eye pressure), *arithmetic means and medians* are calculated to provide a single data point for each patient and each dimension for comparison. To see the distribution for the data dimension for all incidents, the occurrence of the values is counted, providing the *absolute frequency*, and set in relation to the total number of values for that dimension for the *relative frequency*. In order to support the user task to judge the overall development of the visual acuity values for a patient over time, we use the *linear regression function*, setting the visual acuity value as the dependent variable Y (logMar value) and the measurement index as the independent variable X:

  $$Y = \beta_0 + \beta_1 X \tag{5.3}$$

This linear function has β_0 as the intercept and β_1 as the slope. To derive β_0 and β_1, we set m =number of measurements for a patient and use the least squares estimate:

$$\beta_0 = \frac{(\sum y_n)(\Sigma x_n^2) - (\Sigma x_n)(\Sigma x_n y_n)}{m(\Sigma x_n^2) - (\Sigma x_n)^2} \tag{5.4}$$

and

$$\beta_1 = \frac{m(\Sigma x_n y_n) - (\Sigma x_n)(\Sigma y_n)}{m(\Sigma x^2) - (\Sigma x)^2} \tag{5.5}$$

with $n = 0..m$.

Using the visual acuity values for $x_{0..m}$ and their indices as $y_{0..m}$, this function produces a very precise, yet approximative, visual acuity change for each patient over the treatment period. This precision is not necessary to get an impression on the overall performances. After discussion with the experts, we reduce the potentially misleading accuracy of the outcome in two ways. On the one hand, we use the slope results to visually indicate the qualitative development of the visual acuity for a patient, which will be detailed in Section 5.1.3. On the other hand, we use the classes, "gain," "unchanged" and "loss," from the ophthalmic domain (see Wecker et al. [Wec+17]) with the following boundaries:

$$f(x) = \begin{cases} \text{loss} & : & & \Delta V_{letter} & < & -15 \\ \text{unchanged} & : & -15 & \leq \Delta V_{letter} & \leq & 15 \\ \text{gain} & : & 15 & < \Delta V_{letter} & & \end{cases}$$

These classes allow users to mentally assign patients into a performer group and aim at preserving the expressiveness of the regression result by reducing the precision.

- **Sorting of patient data:**

  The sorting function uses different data dimensions to rearrange the order of patients. It ranges from simple ordering of patients by numerical values up to more complex functions like grouping patients first and sorting within this group.

5.1.3 User Interface Layer

For the user interface layer, we follow the common approach to separate between visualization and interaction, each representing an own component.

Visualization Component

To follow our user-in-the loop workflow introduced in Section 3.3, we created three screens for the visualization component. The first two are the data import/export

and preprocessing screens and the third is the data cleansing and exploration screen. Both the *data import/export* and *data preprocessing* screens (Figure 5.2) have been

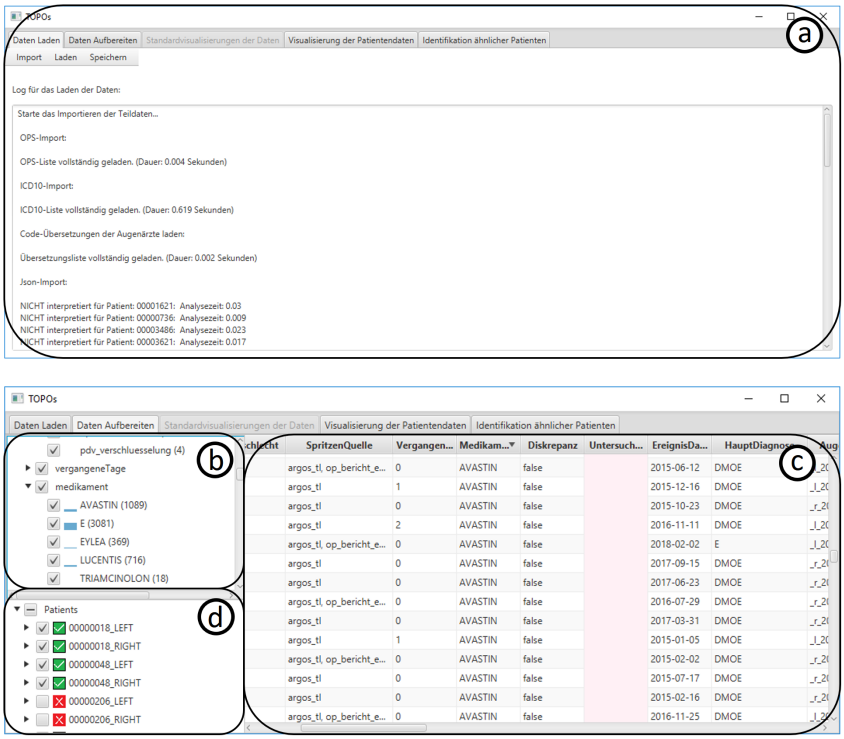

Figure 5.2: Data import/export screen with log information (a). Data preprocessing screen (b) through (d). (b) the distribution view, showing the distributions for all data dimensions of the incidents, including the absolute frequency and relative frequency. (c) the incident view, showing all dimensions for one incident. (d) the patient view, showing the patient-oriented structure to show all information for one patient.

designed in line with the users' mental map of data handling in the clinical environment. They are text and table based with minor visual elements. This allows the users to identify themselves with the tool and the data within. We visually separated the data import/export from the data preprocessing, so that users are aware that there is a clear distinction between the external data-source retrieval and further internal data processing. The actual visual analytics process starts, after the original data is successfully imported. The content of the data import/export screen (Figure 5.2 (a)), is a user menu to support the import/export interactions and a log information view displaying feedback information on for data import, export, and parsing.

The data preprocessing screen (Figure 5.2 (b) through (d)) is split into three views, which represent different aspects of the data. On the top left side of the screen, the values for each incident dimension are shown in a tree view (b). The tree

view allows to hide/show relevant dimensions with their values and additionally the absolute and relative frequency, giving clinicians the possibility to find obvious outliers and main contributors for one dimension. On the lower left side of the screen, the patient view is located (d). It is organized as a tree view providing detailed text-based information for each patient in node form. By that, patients whose time series contain no relevant information, can be detected. Each patient node visually indicates if the patient is included in the dataset for visualization. This allows users to quickly judge if all empty patients have been excluded from the visualization. The incident view on the right side (c) that shows all incidents of the data vertically stacked in a table, has the major share of the screen. Each table cell represents one dimension of the multivariate incidents. The data is shown in text form, except for empty data cells, which are moderately emphasized to ease the location of largely empty incidents, without distorting the users' attention from the filled data cells.

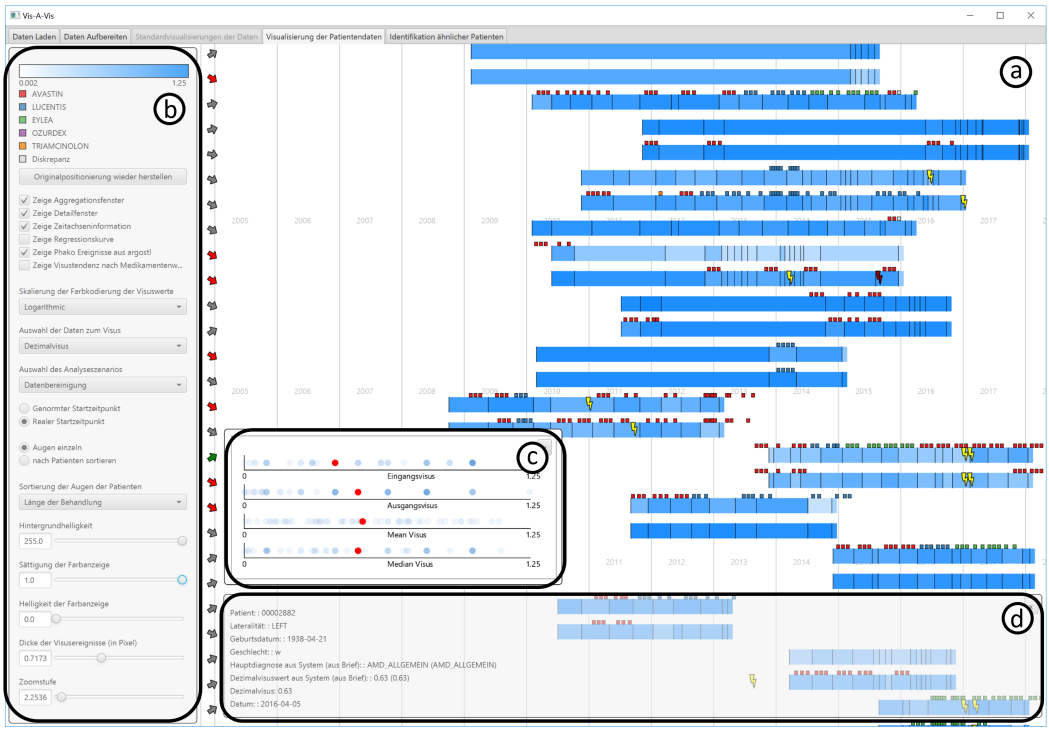

Figure 5.3: The patient data visualization screen. The main visualization view (a) shows the time-oriented data with data segments and incidents. The control panel (a) holds the legend and the parameter setting. The summary panel (c) shows aggregated values for patients and their distributions for all patients. The detail view (d) shows detailed information on a specific data point on demand.

The *data cleansing and exploration* screen shown in Figure 5.3 holds the patient data views. Here, the design is more complex, as the goal is to communicate time-oriented data of different types from multiple sources, with redundancies and potential discrepancies. On top of that, the data derived from the analytics layer has to be integrated. With respect to that, we decided to show the content in four separate views.

Main View (a) The purpose of the main view is to allow data cleansing and explo-

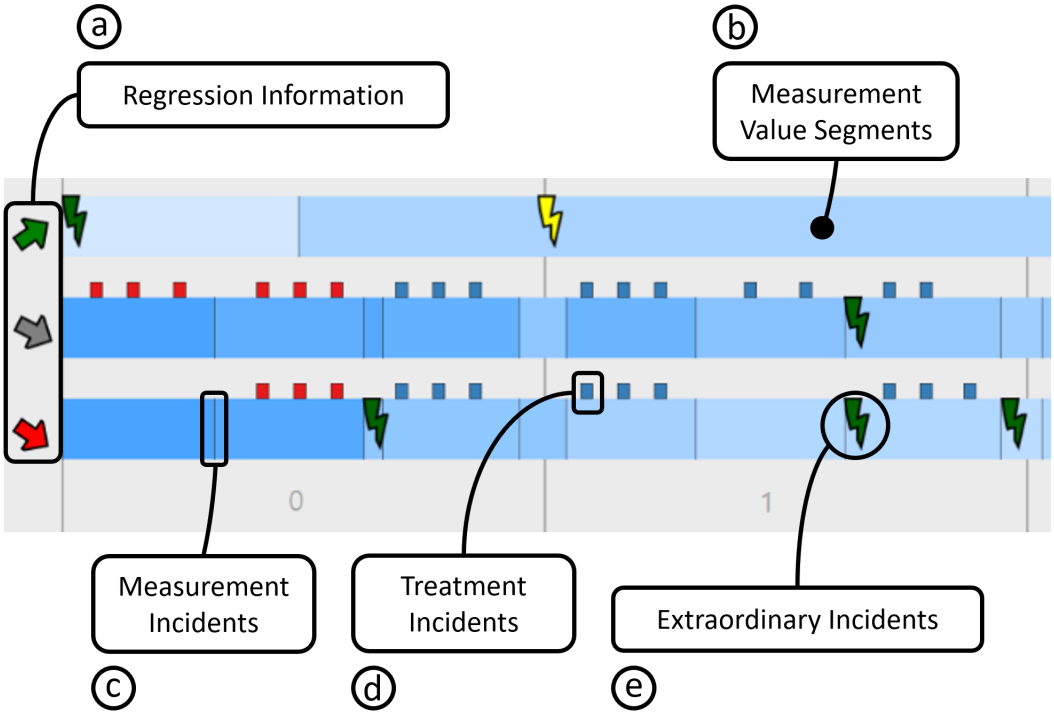

Figure 5.4: Detailed view of the patient data visualization. At the beginning of each horizontal bar the regression slope information is shown (a). The bar itself consists of color-coded segments each representing a visual acuity measurement value (b), the measurement incidents at a certain date (c), the color-coded treatment incidents, showing an injection and the medication used (d). Extraordinary incidents are visualized as color-coded flashes (e).

ration by visualizing the defined patient data dimensions as detailed in Figure 5.4. As the visualized data dimensions are segments and incidents in time for different patients, we divided the visualization space into horizontal rows, which evolve in time from left to right. Each row represents the data for one eye of a patient. The rows are divided into segments (b) that represent a certain time range. They are encoded as color-coded rectangles. The color communicates either visual acuity

values or visual acuity deviation using appropriate color schemes as published in Harrower and Brewer [HB03].

Incidents (c), (d), & (e), on the other hand, refer to a specific point in time and are therefore represented as small glyphs. They can be either regular (c) & (d) or extraordinary (e). Since they have varying medical consequences, it is important to differentiate between them.

For this reason, we designed two types of glyphs. The regular incident glyphs represent regular incidents as rectangles and lines, smoothly connected to the respective horizontal bar. Glyphs that represent extraordinary incidents, on the other hand, have an irregular shape, to show the disconnectedness from the regular data. To allow users to draw conclusions from the different data dimension visualized, the locations of all time-oriented glyphs on the screen refer to the actual point in time of this data point. Besides incidents, the main view also communicates information from the analytic layer. For example, to show the regression loop for a patient, we designed additional glyphs and located them separated in front of each time-oriented bar to refer them to a specific patient. As they communicate aggregated data, which does not refer to a particular point in time, we show them disconnect from the time-oriented data.

The Control Panel (b) supports steering and understanding the main view. It is designed to provide an explanatory legend for colors used in the main view. Additionally, it holds parameter controls to display the current parameter state and allow parameter adjustment by the user. As we have different parameter types, such as numerical, categorical, or boolean, we also use different control elements like check-boxes, slides or drop-down boxes. **The Summary View (c)** shows aggregated data for four data dimensions (first, last, mean, and median visual acuity values) for all patients at once. In contrast to the main view, which shows time segments, here the data per time point is displayed. The color intensity encodes the frequency of associated data points.

Finally, **the Detail View (d)** provides detailed information for just one data point for a patient. Whereas the summary allows for a general overview over all patients, the detail view supports the judgment of a single data point for cleansing or exploration.

Interaction Component

For an appropriate implementation of the given workflow, various interaction techniques are required to complement the visualization design.

For the views on the *data import/export screen* and *data preprocessing screen* (Figure 5.2), the interaction techniques are designed in accordance with existing ones in the clinical system, in order to fit the domain experts' expectations. The data preparation views enable the user to browse through the data and possibly select

relevant or filter out non-relevant patient data. We support this by interaction methods like scrolling, row selection, sorting, column rearrangement as well as filtering via mouse interaction.

For the views on the visualization screen (Figure 5.3), interaction methods get more complex, as they have to support (i) the visualization parameter adaptation, (ii) the visualization navigation, (iii) the data cleansing, and the (iv) data annotation. The visualization parameter adaptation can be performed via the dedicated parameter control panel described above. The user can change specific parameters with their controls or choose preset parameter settings using a combobox. By that, the visualization is changed on the fly. Scrolling and zooming allows to navigate through the visualization, to find data points for cleansing or interest of exploration. Selecting such points through mouse hovering provides additional information on demand. The third interaction goal is the cleansing support, which has to take care of marking and/or removing of existing data points and the creation of new ones.

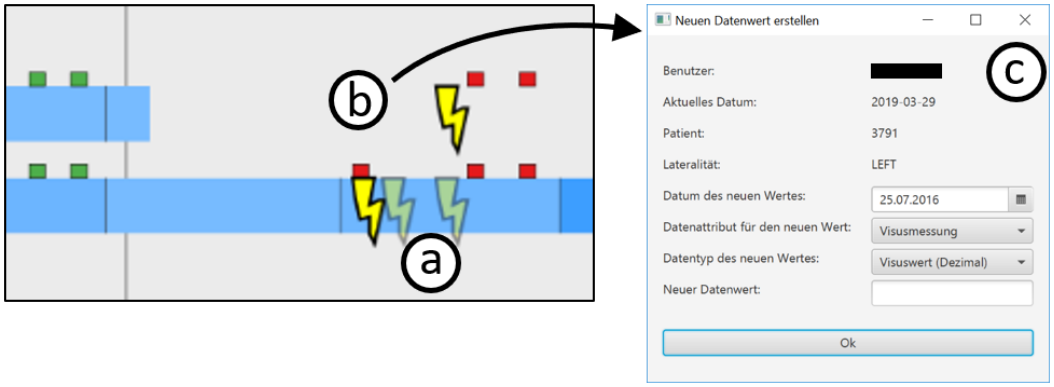

Figure 5.5: Interaction support for cleansing. Segments or incidents can be cleansed with a mouse click (a). New incidents can be created in the visualization (b) support by an interactive dialogue (c).

For this purpose, we introduce cleansing functions as shown in Figure 5.5, that either mark a specific data point as “cleansed” using transparency (a) or allow the creation of new data points with local reference in the visualization (b) supported by an interactive dialogue (c).

Finally, the interaction must support annotation creation to mark and document findings or insights during the exploration loop. This can be done with the annotation function as shown in Figure 5.6. Similar to the cleansing, a data point is marked (a). Yet here it is not changed, but can be amended with a comment, holding information from the user. This comment is entered in a separate dialogue (b). All generated comments for that data point are displayed in the detail view depicted above.

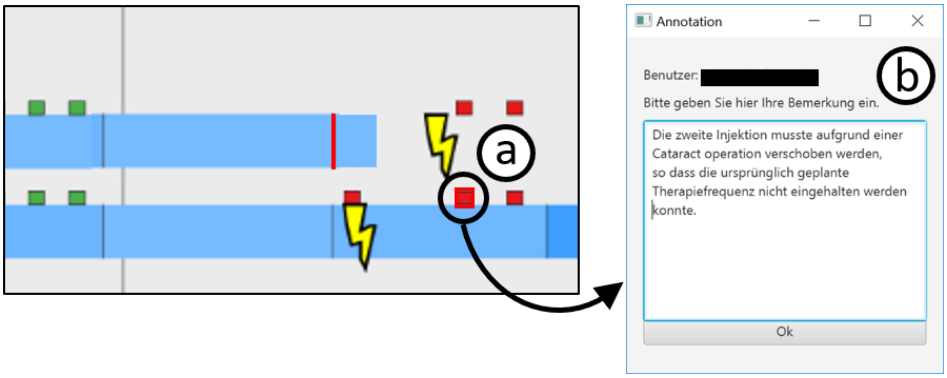

Figure 5.6: Interaction methods for data annotation during exploration. A data point is marked with a mouse click interaction (a). This opens the annotation dialogue (b).

5.2 Implementation of Annotations

In this section, we describe the implementation of the annotation functionality into the modular tool from Section 5.1. We start with the assessment of the implementation requirements of the annotation functionality. This is done in respect to the modules in the existing tool. On the basis of the assessment results, we design our annotation structure and implementation concept. Finally, we implement this concept into the existing software.

5.2.1 Implementation Requirements for Annotation Functionality

In order to derive the implementation requirements for our annotation functionality, we apply Munzner's nested model as it is a convenient method to tackle these problems ([Mun09]). At this point, we have the annotation requirements, the annotation characteristics, the annotation design, the existing tool design, the existing tool functionality, and the existing tool architecture. Hereafter, we sort this information into the nested model, and thus derive the necessary information for the annotation implementation requirements.

Domain Problem Characterization

For the domain problem characterization, we must learn about the tasks and data of target users in our particular target domain of annotations for heterogeneous clinical data in the field of ophthalmology. With the help of our annotation characterization model, we have identified the use case requirements and visual analytics requirements for the different steps in the visual analysis in Section 3.3. Besides the general annotation requirements for each step, we also detailed the use case and visual analytics requirements for medical data in the medical domain in Chapter 4.

Data/Operation Abstraction Design

According to Munzner, the abstraction stage is to map problems and data from the vocabulary of the specific domain into a more abstract and generic description that is in the vocabulary of computer science. We achieve this with the help of the use case and visual analytics requirements and our annotation characteristics model, as we derived the respective annotation characteristics in the different steps in general and for medical data in particular within the Sections 3.3, and 4.2 thru 4.4.

Encoding Interaction Technique Design

The third level is designing the visual encoding and interaction, which was done in Chapter 4 of this work. Here, we develop a fitting annotation design on the basis of the suitable annotation characteristics.

Algorithm Design

Finally, in accordance with the nested model, the innermost level is to create an algorithm to carry out the visual encoding and interaction designs automatically. The basis for this algorithm is formed by the implementation requirements, which ensure that the needs of annotation implementation can be respected when extending the tool functionality with annotation functions.

The first implementation requirement concerns the annotation data, which needs to be integrated into the visual analytics system. It is important that the annotation data is stored separately from the original data, to avoid an unwanted altering or amendment of the original data. Nevertheless, there need to be links between the original data and related annotation data, e.g., for data preprocessing annotations, which concern specific consolidated data points. Therefore, the first implementation requirement is an independent annotation data structure with the possibility to be linked to one or more data points of the original data.

The second implementation requirement concerns the implementation of the visualization and interaction techniques for annotations. Two things are of importance. On the one hand, the visualization techniques need to support the annotation design and its purpose for the specific annotation. On the other hand, the annotation design implementation must fit the underlaying data visualization in a way, that both the data visualization and the annotation goals are respected. We can reach this if we refer to the varying annotation characteristics. For example, the implementation requirement for annotations with the characteristic “visual separation” is that the implementation ensures two distinct visualization areas for the original data visualization and annotation visualization.

The third and final implementation requirement concerns the annotation management. The goal is to allow for recording, storage, retrieving, and/or externalization

the annotations as well as the managing of the interplay between the original data and annotations. Here, we need to define algorithms that enable (i) automatic internal mechanisms to connect annotation data with original data, (ii) decide on the combined interaction techniques, e.g., recording of editing and annotation actions during data cleansing, and (iii) manage the visualization of both the original data and annotations.

5.2.2 Implementation of the Annotation Functionality

With the implementation requirements at hand, we perform the innermost step in the nested model, the implementation of the system. As the underlying tool is a modularized solution with suitable interfaces, we can add our functionality by extending this architecture. We amend the existing units and integrate an additional annotation management unit. The extended architecture of the tool is shown in Figure 5.7. The added annotation unit and amended annotation functionality is shown in orange, the existing units are purple, and the existing functionality is green.

Extended Data Unit

Our extensions in the *data unit* encompass the setup and processing of the annotation structures, including the interplay with the original data (Figure 5.7 lower left). To distinguish between data and annotations, we set up two additional internal data structures, one for data linked annotations and one for visualization linked annotations. The added annotation structure contains all necessary data for the annotation, such as

- timestamp,
- user,
- annotation content (category, free text, etc.),
- reference to original data either on data point value, data point, data point group (e.g., all data points for one patient), or visualization area level.

With the help of this structure, we fulfill the first implementation requirement and allow for data recording and linking to original data and/or visualization without changing the original data.

Added Annotation Management Unit

The *added annotation management unit* (Figure 5.7 middle left) is responsible for the annotation management within the system. It receives and structures the annotation information from the user interface unit and sends it to the data unit

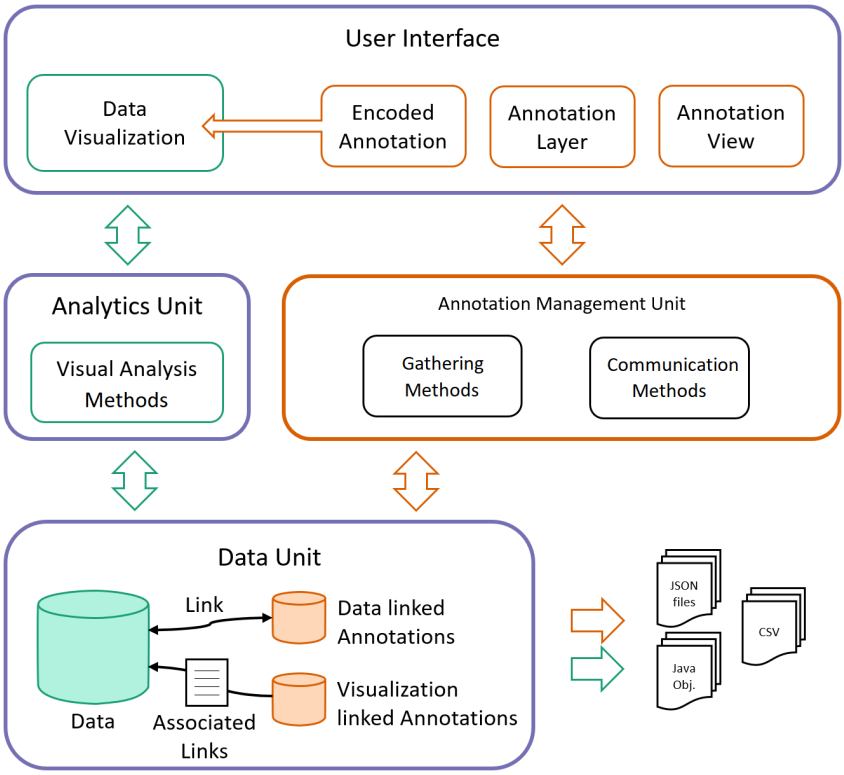

Figure 5.7: The extended high-level architecture of the tool with additional annotation functionality. The original architecture units are purple, the original data and algorithms are green. The added annotation information and annotation algorithms are brown.

for storage. Conversely, it requests the necessary information from the data management unit and forwards it to the user interface unit or to the file system for externalization. The developed algorithms automatically record and structure the necessary information for the annotations either during the internal data preprocessing actions or via the user interface thru user interaction during data cleansing and data exploration. This respects the third implementation requirement.

Extended User Interface Unit

The extended *user interface unit* (Figure 5.7 top) contains the original data visualization functionality together with additional screen management functions. These ensure the appearance of the annotations dependent on their characteristics and the current step in the analysis. For data preprocessing, it allows the merging of the data and annotation visualization to enable visual encoding. For the data cleansing visualization, it implements a second layer on the original data visualization. This layer contains the annotation visualization, which overlays the original data, and

thus allows for local highlighting or overlaying the original data. It can be switched on/off via user interaction. For data exploration, the user interface unit creates an additional layer on top of the original data view to allow annotation visualization for highlighting purposes. Additionally, the user interface unit creates a separate annotation view, which contains the user comment section. Due to these combinations of data visualization and annotation visualization, the user interface unit fulfills the third implementation requirement.

In order to support the annotation interaction functionality, we extend the user interface unit by adding various interaction means. First, we implement additional annotation interaction functionality for recording, which depends on the current annotation task. During data preprocessing, the annotation creation for consolidation is done fully automatic, so the only interaction function needed, is for clarification of discrepancies (e.g., accept automatically derived result or discard data point). For this purpose, we add several mouse events. For data cleansing, for example, we use the existing data cleansing interaction functions, e.g., right mouse click to edit a data point value, and extend the resulting editing screen with annotation fields. These fields can be filled via user interaction (e.g., radio button selection via mouse click or keyboard use for text fields). During data exploration, the interaction functionality combines typical data exploration interactions such as browsing, zooming, and scrolling, with annotation interaction such as mark and comment. The user interface unit supports this by integrating the annotation interaction functionality into the data exploration functionality based on the screen views. In this way, the original data view with the annotation layer supports additional mouse events, such as right click to set marks, while the annotation view with the comment section has dedicated buttons to support the comment, discussion, and externalization function. The externalization function starts an externalization algorithm within the annotation management unit that converts the internal annotation structure to a user defined structure (JSON, CSV, or Java object) and sends it to the file system.

With this extended tool architecture, we allow a structured annotation recording and communication with the possibility to relate annotations to the original data or data visualization. Furthermore, it is possible to externalize the annotations upon user request.

5.3 Discussion

With the implementation of the annotation functionality into an existing visualization system, we have shown that it is possible to extend a data visualization system with annotation functionality. One problem during implementation may be the fulfillment of an annotation requirement if the original data visualization does not provide the necessary information or data presentation for the annotation needs of the experts. In these cases, either the original data visualization must be changed, which may create high effort, if the data visualization implementation is outdated

or has been implemented by someone else. Another option is the creation of a new holistic system, containing both data visualization and annotation functionality. In our case of visualizing heterogeneous clinical data, we could sufficiently change the data visualization, in order to meet the goals of annotation. This is due to the same specialists for both the visualization system and annotation extension, as well as the modular architecture of the software. This enabled a structured integration of new modules containing the annotation functionality as well as extending the existing modules. Another reason for convenient annotation implementation is the straightforward approach of the original data visualization. Due to the limited number of used visualization techniques, a sufficient number of visualization techniques was available for the annotations.

6 Expert Feedback

In addition to the collaborative development process of the annotation characteristics as well as the system implementation, we used several approaches to gain user feedback. Two major feedback sessions were organized with the domain experts from ophthalmology and based on our visual analytics tool. This will be described in the first section of this chapter. The second section is a short outlook on user feedback concerning the application of our annotation characteristics model to epidemiological data. This concerns data from several clinics, which was gathered from Covid-19 cases.

6.1 Expert Feedback on Annotations for Medical Data

With the annotation tool at hand, we organized two sessions to gather feedback from experts in the field of ophthalmology. These experts are specialized on retinal diseases and come from a clinical eye care center. Two of these experts have been integrated in the visual analytics approach from the beginning as part of a research project.

The first session was designed as an application session with an expert, usually working in the conventional retrieval and analysis of patient data (cf. Section 2.1.2). The second session was held as a tool demonstration with a mixed group of experts, including ophthalmology practitioners and research scientists.

With the *application session*, our general goal was to test the practical utility of the tool. Particularly, we aimed to assess: (i) the appropriateness of the visual design and (ii) the usability of the interaction functionality to solve the identified tasks (cf. Section 2.1). To answer these points, a senior retinal physician applied the tool supported by a visual analytics expert. The data for the session consisted of an arbitrarily chosen sample of 205 patients with a total of 9790 regular and irregular incidents. All workflow steps were performed within a time frame of 60 minutes.

Starting with the data preprocessing, the expert's first objective was to identify any flaws in the data. The preprocessing annotations, representing the import log and preprocessing screens helped to reveal several major issues, including missing or irrelevant values. The expert applied automated filtering rules and used available interaction functions to exclude the corrupted data entries. As a result, a structured dataset with 204 medically relevant patients and 9104 incidents was obtained.

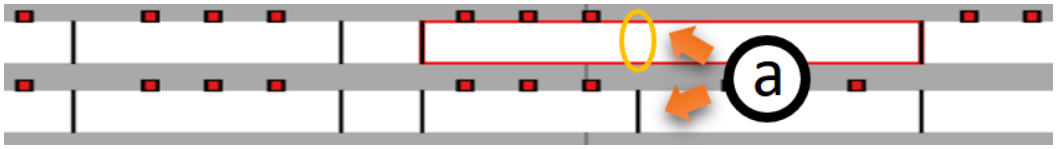

Figure 6.1: The data cleansing task. The user has adjusted the visualization parameters, so that missing visual acuity measurement incidents can be detected. (a) The arrows point at the left and right eye of the same patient. As the clinic personnel always measures both eyes, it is unusual that on eye misses a measurement.

Continuing with the data cleansing, the expert aimed at improving the data quality. By looking at the patient-data screen, the expert quickly noticed few instances of missing visual acuity measurements. The visualization parameters were switched accordingly to put further emphasis on these issues (Figure 6.1). Browsing through the data helped the expert to classify them with as a reoccurring problem, with 73 missing values found in total.

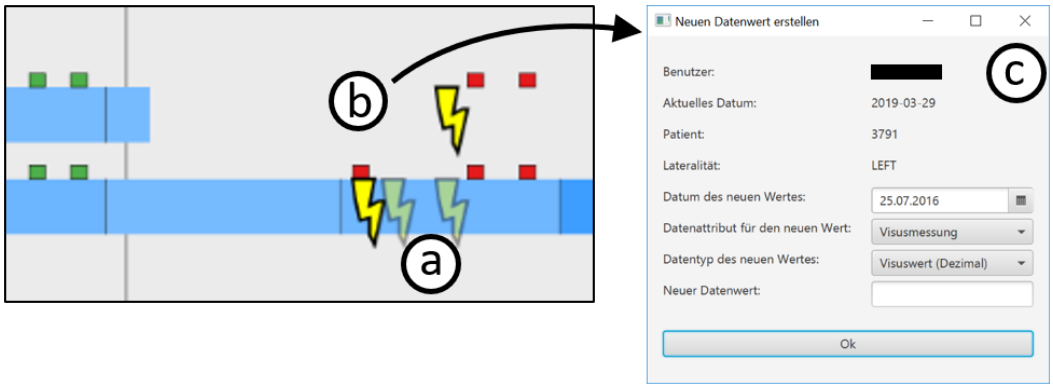

Figure 6.2: Annotation support for data cleansing. Segments or incidents can be cleansed by users with a mouse click (a). New incidents can be created in the visualization (b) support by an interactive dialogue (c).

Using the annotation function (Figure 6.2), all missing values could be directly amended. In this regard, the expert pointed out the interactive manipulation of the visualized data to be particular intuitive, useful, and time efficient with the integrated annotation recording. Finally, the expert explored the cleansed data to study the visual acuity development in relation to injections received. An appropriate parameter preset for the patient-data screen was loaded and refined. While browsing through the data, the expert located several extraordinary incidents, such as a patient with a sudden loss of visual ability. The linked summary and detail views allowed to investigate the incidents, check their plausibility, and record gained insights. The expert reassured us that the provided functionality was indeed helpful

to draw the conclusion that such incidents are unexpectedly common and strongly influence the visual acuity development. In the end, all tasks of the application session were completed successfully, including the annotation of exploration results and comment recording. In retrospect, the expert particularly appreciated the ability to jump back and forth between the exploration and cleansing loops to immediately process every discovered erroneous data point. The expert concluded that reducing the manual data processing effort compared to current procedures while eventually being able to obtain analysis results with higher accuracy are great benefits.

With the *demonstration session*, our general goal was to assess the medical relevance of the design concept. A group of three ophthalmic experts, including a head physician, from an eye care center specialized in the diagnosis and treatment of retinal diseases participated. A live demonstration of the tool and its main components was given by a visual analytics expert based on the described data and workflow. Informal feedback was gathered during the demonstration and in subsequent discussions. The experts particularly appreciated the workflow for enabling a more structured way of working with the clinical data. They also considered the unified access to data preprocessing, data cleansing, and data exploration as well as the developed interplay of those components to be highly meaningful for improving the quality of the data and the analysis results. Regarding the design, one expert stated: "After looking at the visualization for a while, I begin to recognize patterns, similar to looking at patient images." In the discussions, this statement was attributed to the new visualization design choices (row-wise arrangement of patient data and applied color presets) as well as to the consideration of familiar presentations known from the conventional data analysis. Overall, the feedback of the second session was very positive. The physicians appreciated the workflow for its structured appearance and the design for its combination of familiar designed views and new approaches for the field of ophthalmology with high functionality. The experts also rated the design concept with annotation function to be effective and the medical outcome to be highly relevant. Based on the demonstration, the experts even decided to present the tool and the generated results at the largest ophthalmology congress in Germany, the DOG Congress in Bonn 2018 [Eth18].

6.2 Outlook on Annotations for Epidemiological Data

In order to test the annotation characterization model on other data, we conducted a preliminary annotation problem solving approach on epidemiological data. We conducted a short requirements engineering approach consisting of several interviews with epidemiological experts, to find both the data and annotation requirements. The result was that the focus lays on the development of the disease over time for the original data. Concerning the use case requirements for the annotations, we learned that the annotations should allow to collect doctoral reports on specific Covid-19 cases on patient level. The doctoral reports contain the judgement,

examination, and conclusions from the original data visualization. Based on these requirements, we applied our annotation characterization model. As stated, the use case requirement is the reporting of patient related Covid-19 development issues. The visual analytics requirements focus on the need to combine the development with the extracted insights and findings from the doctoral analysis to create the report. This leads to the following annotation characteristics:

- What are annotations?
  - Free Text
  - *Reason:* With the help of free text, the users can freely integrate their conclusions into the report within the system.
  - Graphical Items
  - *Reason:* The graphical items allow the highlighting of the annotation location.
- Why do we annotate?
  - Add User Information
  - Add Outcome Information
  - *Reason:* The explanations of the experts are a combination of existing knowledge within their minds and the insight from the visualized data.
- How to Gather Annotations?
  - Alphanumerical Input
  - *Reason:* Report needs to be readable text from experts.
  - Marks
  - *Reason:* Marks are used to locate the annotation reference to the original data.
- How to Communicate Annotations?
  - Visual Separation
  - *Reason:* Clear distinction between data and report is needed to allow experts to differentiate between data and report. If a shared border between the views is supported, the local reference to the data is possible.

Based on the visualization requirements and annotation characteristics, we designed a draft for a holistic approach for both the data and annotation visualization and interaction. The draft is shown in Figure 6.3. On the left side, a vertical timeline with encoded examination results from the original data is located. This is inspired by the LifeLines approach from Plaisant et al. [Pla+98]. The annotations are located on the right side. The border between the data and the annotation is the

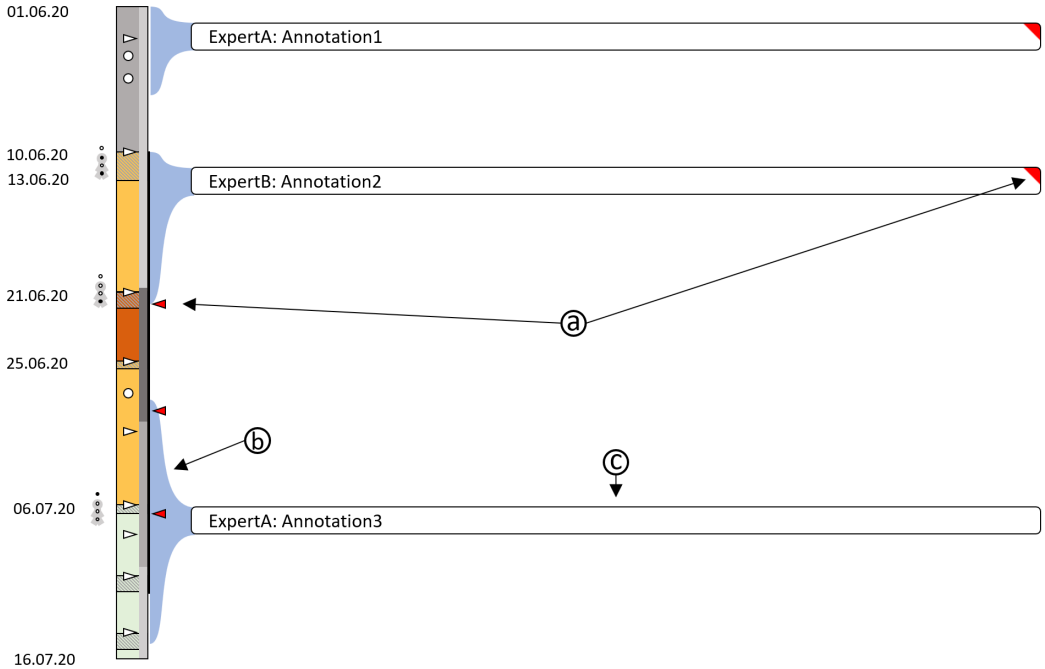

Figure 6.3: Draft of a data and annotation visualization for epidemiological data. The visualization is time-oriented from top (early) to bottom (late) and shows the development of a patients Covid-19 infection. The annotations are shown on the right sight. There are glyphs to mark locations of further annotation detail (a), a marked time period (b) for the text-based annotation comment (c).

vertical line defined by the right border of the time bar and the left border of the blue area marks for the text-based annotations. The annotation functionality is supported by three visualization and interaction elements on the left side of the vertical time bar. First, there are red triangles, indicating an annotation incident (a date on the timeline) and/or further annotation detail (a). Second, there are blue glyphs, linking a time period (in contrast to an incident) of the original timeline to an annotation. Third, the explanatory annotation content from experts is shown in text fields (c).

In a presentation session with experts, this annotation design and functionality was generally appreciated. They commented that it provides a good overview on the patients' data in combination with the experts' comments. They appreciated the mouse over functionality to provide detail information, as this reduces the visual clutter on the one hand and can provide detail information on demand on the other hand. Furthermore, they liked the possibility to validate probable infections via the annotations. Finally, they appreciated the possibility to integrate additional

research information, especially on the highly dynamic development of the Covid-19 research.

As an improvement they demanded an explanation of the used glyphs, colors, and functionality and further data correction functionality, e.g., a data cleansing approach.

7 Conclusion and Future Work

We have shown that a systematic annotation development can improve the overall annotation application in visual analytics. Additionally, a sophisticated workflow for a visual analytics approach on heterogeneous and erroneous clinical data can be beneficial for the analysis. In this chapter we conclude our work and provide an outlook for further research questions to be tackled.

7.1 Conclusion

Within this work, we tackled several research questions on annotations in visual analytics for medical data. We found that different understandings of the term annotations exist. For clarity, we used the following understanding in this work: *Annotations are information within the visual analytics system, which is added during the visual analytics process.*

With this definition at hand, we identified the research questions on which annotation characteristics exist in literature. In addition, we derived, how they can be systematized, and how we can create a structured approach to characterize annotations in visual analytics. We tackled these questions in three steps. In Chapter 2, we conducted a brief literature survey on annotations where we identified various annotation characteristics. In Chapter 3, we (i) developed a morphological box approach wherein we list the identified and sorted annotation characteristics, (ii) developed an annotation characteristics model, and (iii) developed a user-in-the-loop workflow to preprocess, cleanse and explore heterogeneous medical data. In Chapter 4, we developed an approach to suitably design the annotations. In order to judge our approach, we implemented the annotations into an existing tool in Chapter 5 and organized several sessions to receive feedback from experts, which we described in Chapter 6. Furthermore, we briefly tackled epidemiological data in Section 6.2.

In the following section, we discuss the results of our research in the respect to its benefit to the field of annotations in visual analytics.

7.2 Discussion

In the conclusive discussion, we would like to depict how the research community benefits from this work and what limitations exists, which will be further detailed in Section 7.3. To this date, annotations have been widely used to integrate additional knowledge into the analysis. The state of the art depicts annotation principles, describes annotation usage examples, or works with previously recorded annotations. In this respect, many useful annotation characteristics have been developed and applied to support the overall data analysis.

Our work extends this state of the art by providing an organized overview on many of the annotation characteristics, in the form of a morphological box. While this mainly concerns annotation literature from visual analytics approaches, other areas, such as linguistics or neural networks where considered, yet not fully covered. Using the morphological box as a basis allows experts to identify and use annotation characteristics in a sorted and structured way for their own annotation development.

In addition, it is now possible to select suitable annotation characteristics for a specific annotation problem using the annotation model. This model provides the parameters to gather all input needed to choose the annotation characteristics. Furthermore, this model also defines design principles to derive a fitting annotation design for the chosen annotation characteristics. Nonetheless, this model needs to be seen in the context of a particular annotation problem on a particular use case. It does not provide general, use case independent, annotation characteristics.

Finally, this work helps to extend the considerations for some of the steps in visual analytics. A dedicated workflow to approach data preprocessing, data cleansing, and data exploration with integrated annotations, can help visualization experts in tackling visual analytics problems. We have demonstrated this for an approach on heterogeneous clinical data. A brief analysis has shown that broader application, e.g., to data from different clinics, can also be promising. While we have covered some of the steps in visual analytics, other steps, such as knowledge generation from the knowledge generation model from [Sac+14], remain open research.

In this way, we have extended the knowledge in field of annotations in visual analytics, while many research questions remain. Some of these questions will be detailed hereafter.

7.3 Future Work

While we have shown that annotation characteristics can be structured and systematically derived, many questions remain open or arose from our research. In this final section we describe some of the open research questions.

Other Basic Questions: Within our literature survey we were able to identify basic questions on annotation characteristics, which allow a structured characterization. Nonetheless, we did the literature survey on the topic of visualization. Future work could extend the literature survey to other domains, such as linguistic or neural networks. It would be interesting to examine if our questions also apply for these domains.

Other steps in VA: While we examined the visual analytics steps data preprocessing, data cleansing, and data exploration, there are other steps in visual analytics, such as verification and knowledge generation as proposed by Sacha et al. [Sac+14]. Future work could address their overall integration into a data preprocessing, cleansing and exploration loop, to address, e.g., knowledge generation issues. Open questions are, for example, if it is necessary to allow a loopwhise iteration, including data cleansing, during knowledge generation.

Evaluation: While we did conduct a sophisticated feedback session with domain experts, it would be interesting to see the results from a thorough evaluation, such as qualitative and or quantitative studies. With the help of such studies, it could be found, if and how the characteristics, the model, the workflow and/or the visual analytics system can be improved.

Bibliography

[AAU15] R. Alm, M. Aehnelt, and B. Urban. "Processing Manufacturing Knowledge with Ontology-Based Annotations and Cognitive Architectures". In: *Proceedings of the International Conference on Knowledge Technologies and Data-driven Business*. 2015, pp. 1–6. DOI: 10.1145/2809563.2809576.

[Adr+18] S. D. Adrean, S. Chaili, H. Ramkumar, A. Pirouz, and S. Grant. "Consistent Long-Term Therapy of Neovascular Age-Related Macular Degeneration Managed by 50 or More Anti-VEGF Injections Using a Treat-Extend-Stop Protocol". In: *Ophthalmology* 125.7 (2018), pp. 1047–1053. DOI: 10.1016/j.ophtha.2018.01.012.

[Ahn+03] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford. "CAPTCHA: Using Hard AI Problems for Security". In: *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques*. 2003, pp. 294–311. DOI: 10.1007/3-540-39200-9_18.

[Ahn+08] L. von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum. "reCAPTCHA: Human-Based Character Recognition via Web Security Measures". In: *Science* 321.5895 (2008), pp. 1465–1468. DOI: 10.1126/science.1160379.

[Al-+13] A. Al-Naser, M. Rasheed, D. Irving, and J. Brooke. "A Visualization Architecture for Collaborative Analytical and Data Provenance Activities". In: *Proceedings of the International Conference on Information Visualisation*. 2013, pp. 253–262. DOI: 10.1109/IV.2013.34.

[BK53] E. Brunswik and J. Kamiya. "Ecological Cue-Validity of 'Proximity' and of Other Gestalt Factors". In: *The American Journal of Psychology* 66.1 (1953), pp. 20–32. DOI: 10.2307/1417965.

[BL76] I. L. Bailey and J. E. Lovie. "New Design Principles for Visual Acuity Letter Charts". In: *American Journal of Optometry and Physiological Optics* 53.11 (1976), pp. 740–745. DOI: 10.1097/00006324-197611000-00006.

[BLM12] S. B. Bloch, M. Larsen, and I. C. Munch. "Incidence of Legal Blindness From Age-Related Macular Degeneration in Denmark: Year 2000 to 2010". In: *American Journal of Ophthalmology* 153.2 (2012), 209–213.e2. DOI: 10.1016/j.ajo.2011.10.016.

[Bou+17] N. Boukhelifa, M.-E. Perrin, S. Huron, and J. Eagan. "How Data Workers Cope with Uncertainty". In: *Proceedings of the ACM CHI*

Conference on Human Factors in Computing Systems. ACM Press, 2017. DOI: 10.1145/3025453.3025738.

[Bra+14] L. Bradel, C. North, L. House, and S. Leman. "Multi-Model Semantic Interaction for Text Analytics". In: *Proceedings of the IEEE Conference on Visual Analytics Science and Technology*. 2014, pp. 163–172. DOI: 10.1109/VAST.2014.7042492.

[Cal+06] S. P. Callahan, J. Freire, E. Santos, C. E. Scheidegger, C. T. Silva, and H. T. Vo. "VisTrails: Visualization Meets Data Management". In: *Proceedings of the ACM SIGMOD International Conference on Management of Data*. 2006, pp. 745–747. DOI: 10.1145/1142473.1142574.

[CG16] M. Chen and A. Golan. "What May Visualization Processes Optimize?" In: *IEEE Transactions on Visualization and Computer Graphics* 22.12 (2016), pp. 2619–2632. DOI: 10.1109/TVCG.2015.2513410.

[Cha+03] E. Chang, K. Goh, G. Sychay, and G. Wu. "CBSA: Content-Based Soft Annotation for Multimodal Image Retrieval Using Bayes Point Machines". In: *IEEE Transactions on Circuits and Systems for Video Technology* 13.1 (2003), pp. 26–38. DOI: 10.1109/tcsvt.2002.808079.

[CMS99] S. K. Card, J. Mackinlay, and B. Shneiderman. *Readings in Information Visualization: Using Vision to Think*. San Francisco, CA, USA: Morgan Kaufmann, 1999.

[Con+05] A. Conesa, S. Götz, J. M. García-Gómez, J. Terol, M. Talón, and M. Robles. "Blast2GO: a Universal Tool for Annotation, Visualization and Analysis in Functional Genomics Research". In: *Bioinformatics* 21.18 (2005), p. 3674. DOI: 10.1093/bioinformatics/bti610.

[EB12] M. Elias and A. Bezerianos. "Annotating BI Visualization Dashboards". In: *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems*. 2012, pp. 1641–1650. DOI: 10.1145/2207676.2208288.

[Eth18] N. Ether. "Programm DOG 2018". In: *Proceedings of Kongress der Deutschen Ophthalmologischen Gesellschaft*. 2018.

[Fam+97] A. Famili, W.-M. Shen, R. Weber, and E. Simoudis. "Data Preprocessing and Intelligent Data Analysis". In: *Intelligent Data Analysis* 1 (1997), pp. 3–23. DOI: 10.3233/IDA-1997-1102.

[FN13] K. G. Falavarjani and Q. D. Nguyen. "Adverse Events and Complications Associated with Intravitreal Injection of Anti-VEGF Agents: a Review of Literature". In: *Eye* 27.7 (2013), pp. 787–794. DOI: 10.1038/eye.2013.107.

[Fro17] R. Fromont. "Toward a Format-Neutral Annotation Store". In: *Computer Speech & Language* 45 (2017), pp. 348–374. DOI: 10.1016/j.csl.2017.01.004.

[Gru+20] B. Grundel, M.-A. Bernardeau, H. Langner, C. Schmidt, D. Böhringer, M. Ritter, P. Rosenthal, A. Grandjean, S. Schulz, P. Daumke, and

A. Stahl. "Merkmalsextraktion aus klinischen Routinedaten mittels Text-Mining". In: *Der Ophthalmologe* 118.3 (2020), pp. 264–272. DOI: 10.1007/s00347-020-01177-4.

[GS06] D. Groth and K. Streefkerk. "Provenance and Annotation for Visual Exploration Systems". In: *IEEE Transactions on Visualization and Computer Graphics* 12.6 (2006), pp. 1500–1510. DOI: 10.1109/tvcg.2006.101.

[Gsc+12] T. Gschwandtner, J. Gärtner, W. Aigner, and S. Miksch. "A Taxonomy of Dirty Time-Oriented Data". In: *Proceedings of the Multidisciplinary Research and Practice for Information Systems*. 2012, pp. 58–72. DOI: 10.1007/978-3-642-32498-7_5.

[Gsc+14] T. Gschwandtner, W. Aigner, S. Miksch, J. Gärtner, S. Kriglstein, M. Pohl, and N. Suchy. "TimeCleanser: A Visual Analytics Approach for Data Cleansing of Time-Oriented Data". In: *Proceedings of the International Conference on Knowledge Technologies and Data-driven Business*. 2014, pp. 1–8. DOI: 10.1145/2637748.2638423.

[HB03] M. Harrower and C. A. Brewer. "ColorBrewer.org: An Online Tool for Selecting Colour Schemes for Maps". In: *The Cartographic Journal* 40.1 (2003), pp. 27–37. DOI: 10.1179/000870403235002042.

[HS12] J. Heer and B. Shneiderman. "Interactive Dynamics for Visual Analysis". In: *Queue* 10.2 (2012), p. 30. DOI: 10.1145/2133416.2146416.

[HVW07] J. Heer, F. B. Viégas, and M. Wattenberg. "Voyagers and Voyeurs: Supporting Asynchronous Collaborative Information Visualization". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2007, pp. 1029–1038. DOI: 10.1145/1240624.1240781.

[IR04] N. Ide and L. Romary. "International Standard for a Linguistic Annotation Framework". In: *Natural Language Engineering* 10.3-4 (2004), pp. 211–225. DOI: 10.1017/s135132490400350x.

[Jin+17] Y. Jin, J. Li, D. Ma, X. Guo, and H. Yu. "A Semi-Automatic Annotation Technology for Traffic Scene Image Labeling Based on Deep Learning Preprocessing". In: *Proceedings of the IEEE International Conference on Computational Science and Engineering and IEEE International Conferences on Embedded and Ubiquitous Computing*. 2017, pp. 315–320. DOI: 10.1109/CSE-EUC.2017.63.

[Kei+08] D. A. Keim, F. Mansmann, J. Schneidewind, J. Thomas, and H. Ziegler. *Visual Analytics: Scope and challenges*. Berlin, Heidelberg, Germany: Springer, 2008.

[Kei+10] D. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. *Mastering the Information Age Solving Problems with Visual Analytics*. Goslar, Germany: Eurographics Association, 2010.

[KL05] E. Klien and M. Lutz. "The Role of Spatial Relations in Automating the Semantic Annotation of Geodata". In: *Proceedings of the Spatial Information Theory*. 2005, pp. 133–148. DOI: 10.1007/11556114_9.

[Krü+15] R. Krüger, D. Herr, F. Haag, and T. Ertl. "Inspector Gadget: Integrating Data Preprocessing and Orchestration in the Visual Analysis Loop". In: *Proceedings of the EuroVis Workshop on Visual Analytics*. 2015, pp. 7–11. DOI: 10.2312/eurova.20151096.

[Lak+18] K. Lakiotaki, N. Vorniotakis, M. Tsagris, G. Georgakopoulos, and I. Tsamardinos. "BioDataome: a Collection of Uniformly Preprocessed and Automatically Annotated Datasets for Data-Driven Biology". In: *Database* 2018.2018 (2018), bay011. DOI: 10.1093/database/bay011.

[Lin+23] H. Lin, D. Akbaba, M. Meyer, and A. Lex. "Data Hunches: Incorporating Personal Knowledge into Visualizations". In: *IEEE Transactions on Visualization and Computer Graphics* 29.1 (2023), pp. 504–514. DOI: 10.1109/tvcg.2022.3209451.

[Lip+10] H. R. Lipford, F. Stukes, W. Dou, M. E. Hawkins, and R. Chang. "Helping Users Recall their Reasoning Process". In: *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*. 2010, pp. 187–194. DOI: 10.1109/VAST.2010.5653598.

[MF05] H. Mueller and J.-C. Freytag. *Problems, Methods, and Challenges in Comprehensive Data Cleansing*. 2005.

[MGM18] N. McCurdy, J. Gerdes, and M. Meyer. "A Framework for Externalizing Implicit Error Using Visualization". In: *IEEE Transactions on Visualization and Computer Graphics* 25.1 (2018), pp. 925–935. DOI: 10.1109/tvcg.2018.2864913.

[MST12] N. Mahyar, A. Sarvghad, and M. Tory. "Note-Taking in Co-Located Collaborative Visual Analytics: Analysis of an Observational Study". In: *Information Visualization* 11.3 (2012), pp. 190–204. DOI: 10.1177/1473871611433713.

[MT14] N. Mahyar and M. Tory. "Supporting Communication and Coordination in Collaborative Sensemaking". In: *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), pp. 1633–1642. DOI: 10.1109/TVCG.2014.2346573.

[Muñ+00] B. Muñoz, S. K. West, G. S. Rubin, and et al. "Causes of Blindness and Visual Impairment in a Population of Older Americans: The Salisbury Eye Evaluation Study". In: *Archives of Ophthalmology* 118.6 (2000), pp. 819–825. DOI: 10.1001/archopht.118.6.819.

[Mun09] T. Munzner. "A Nested Model for Visualization Design and Validation". In: *IEEE Transactions on Visualization and Computer Graphics* 15.6 (2009), pp. 921–928. DOI: 10.1109/TVCG.2009.111.

[NS10] B. P. Nicholson and A. P. Schachat. "A Review of Clinical Trials of Anti-VEGF Agents for Diabetic Retinopathy". In: *Graefe's Archive for Clinical and Experimental Ophthalmology* 248.7 (2010), pp. 915–930. DOI: 10.1007/s00417-010-1315-z.

[Pla+98] C. Plaisant, R. Mushlin, A. Snyder, J. Li, D. Heller, B. Shneiderman, and K. P. Colorado. "LifeLines: Using Visualization to Enhance Navigation and Analysis of Patient Records". In: *Proceedings of the Amer-*

ican Medical Informatic Association Annual Fall Symposium. 1998, pp. 76–80. DOI: 10.1016/B978-155860915-0/50038-X.

[Rag+16] E. D. Ragan, A. Endert, J. Sanyal, and J. Chen. "Characterizing Provenance in Visualization and Data Analysis: An Organizational Framework of Provenance Types and Purposes". In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), pp. 31–40. DOI: 10.1109/TVCG.2015.2467551.

[Röh+17] M. Röhlig, P. Rosenthal, C. Schmidt, H. Schumann, and O. Stachs. "Visual Analysis of Optical Coherence Tomography Data in Ophthalmology". In: *Proceedings of the EuroVis Workshop on Visual Analytics*. 2017, pp. 37–41. DOI: 10.2312/eurova.20171117.

[Röh+18] M. Röhlig, C. Schmidt, R. K. Prakasam, P. Rosenthal, H. Schumann, and O. Stachs. "Visual Analysis of Retinal Changes with Optical Coherence Tomography". In: *The Visual Computer* 34.9 (2018), pp. 1209–1224. DOI: 10.1007/s00371-018-1486-x.

[Röh+19a] M. Röhlig, R. K. Prakasam, J. Stüwe, C. Schmidt, O. Stachs, and H. Schumann. "Enhanced Grid-Based Visual Analysis of Retinal Layer Thickness with Optical Coherence Tomography". In: *MDPI Information* 10.9 (2019), 266:1–266:23. DOI: 10.3390/info10090266.

[Röh+19b] M. Röhlig, C. Schmidt, R. K. Prakasam, O. Stachs, and H. Schumann. "Towards Accurate Visualization and Measurement of Localized Changes in Intraretinal Layer Thickness". In: *Proceedings of the IEEE Workshop on Visual Analytics in Healthcare*. 2019, pp. 58–59. DOI: 10.1109/vahc47919.2019.8945028.

[Röh+19c] M. Röhlig, J. Stüwe, C. Schmidt, R. Prakasam, O. Stachs, and H. Schumann. "Grid-Based Exploration of OCT Thickness Data of Intraretinal Layers". In: *Proceedings of VISIGRAPP*. 2019, pp. 129–140. DOI: 10.5220/0007580001290140.

[RSS16] M. Röhlig, O. Stachs, and H. Schumann. "Detection of Diabetic Neuropathy - Can Visual Analytics Methods Really Help in Practice?" In: *Proceedings of the EuroVis Workshop on Reproducibility, Verification, and Validation in Visualization*. 2016, pp. 19–21. DOI: 10.2312/eurorv3.20161111.

[Sac+14] D. Sacha, A. Stoffel, F. Stoffel, B. C. Kwon, G. Ellis, and D. A. Keim. "Knowledge Generation Model for Visual Analytics". In: *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), pp. 1604–1613. DOI: 10.1109/TVCG.2014.2346481.

[Sch+19] C. Schmidt, M. Röhlig, B. Grundel, P. Daumke, M. Ritter, A. Stahl, P. Rosenthal, and H. Schumann. "Combining Visual Cleansing and Exploration for Clinical Data". In: *Proceedings of the IEEE Workshop on Visual Analytics in Healthcare*. 2019, pp. 25–32. DOI: 10.1109/VAHC47919.2019.8945034.

[Sch+21] C. Schmidt, B. Grundel, H. Schumann, and P. Rosenthal. "Annotations in Different Steps of Visual Analytics". In: *Proceedings of VISIGRAPP*. 2021, pp. 155–163. DOI: 10.5220/0010198001550163.

[Sno+08] R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng. "Cheap and Fast - but is It Good?: Evaluating Non-Expert Annotations for Natural Language Tasks". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 2008, pp. 254–263. DOI: 10.5555/1613715.1613751.

[SRS18] C. Schmidt, P. Rosenthal, and H. Schumann. "Annotations as a Support for Knowledge Generation - Supporting Visual Analytics in the Field of Ophthalmology". In: *Proceedings of VISIGRAPP*. 2018, pp. 264–272. DOI: 10.5220/0006615902640272.

[SRS20] C. Schmidt, P. Rosenthal, and H. Schumann. *Varying Annotations in the Steps of the Visual Analysis*. 2020. DOI: 10.48550/arXiv.2008.08806.

[SW15] K. M. Shabana and J. Wilson. "A Novel Method for Automatic Discovery, Annotation and Interactive Visualization of Prominent Clusters in Mobile Subscriber Datasets". In: *Proceedings of the IEEE International Conference on Research Challenges in Information Science*. 2015, pp. 127–132. DOI: 10.1109/RCIS.2015.7128872.

[Van+18] P. Vanhulst, F. Évéquoz, R. Tuor, and D. Lalanne. "Designing a Classification for User-authored Annotations in Data Visualization". In: *Proceedings of VISIGRAPP*. 2018, pp. 85–96. DOI: 10.5220/0006613700850096.

[Wal+10] M. Waldner, W. Puff, A. Lex, M. Streit, and D. Schmalstieg. "Visual Links accross Applications". In: *Proceedings of Graphics Interface*. 2010, pp. 129–136. DOI: 10.5555/1839214.1839238.

[Wec+17] T. Wecker, C. Ehlken, A. Bühler, C. Lange, H. Agostini, D. Böhringer, and A. Stahl. "Five-Year Visual Acuity Outcomes and Injection Patterns in Patients with Pro-Re-Nata Treatments for AMD, DME, RVO and Myopic CNV". In: *British Journal of Ophthalmology* 101.3 (2017), pp. 353–359. DOI: 10.1136/bjophthalmol-2016-308668.

[WH11] M. Wang and X.-S. Hua. "Active Learning in Multimedia Annotation and Retrieval: A Survey". In: *ACM Transactions on Intelligent Systems and Technology* 2.2 (2011), 10:1–10:21. DOI: 10.1145/1899412.1899414.

[Wil+11] W. Willett, J. Heer, J. Hellerstein, and M. Agrawala. "CommentSpace: Structured Support for Collaborative Visual Analysis". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2011, pp. 3131–3140. DOI: 10.1145/1978942.1979407.

[Zha+17] J. Zhao, M. Glueck, S. Breslav, F. Chevalier, and A. Khan. "Annotation Graphs: A Graph-Based Visualization for Meta-Analysis of Data Based on User-Authored Annotations". In: *IEEE Transactions on Vi-*

sualization and Computer Graphics 23.1 (2017), pp. 261–270. DOI: 10.1109/TVCG.2016.2598543.

[Zha+18] J. Zhao, M. Glueck, P. Isenberg, F. Chevalier, and A. Khan. "Supporting Handoff in Asynchronous Collaborative Sensemaking Using Knowledge-Transfer Graphs". In: *IEEE Transactions on Visualization and Computer Graphics* 24.1 (2018), pp. 340–350. DOI: 10.1109/tvcg.2017.2745279.

[Zwi67] F. Zwicky. "The Morphological Approach to Discovery, Invention, Research and Construction". In: *New Methods of Thought and Procedure*. Berlin, Heidelberg, Germany: Springer Berlin Heidelberg, 1967, pp. 273–297. DOI: 10.1007/978-3-642-87617-2_14.

Curriculum Vitae

Personal Data

Name:	Christoph Schmidt
Wohnort:	Schwerin
geboren am:	09.07.1978
geboren in:	Rostock
Nationalität:	deutsch

Scientific Career

2017 - 2021	Wissenschaftlicher Mitarbeiter am Lehrstuhl für Computergrafik, Universität Rostock
2017 - 2021	Promotionsstudent am Lehrstuhl für Computergrafik, Universität Rostock
2000 - 2007	Studium der Multimediatechnik, Hochschule Wismar
1998	Abitur, Sprachgymnasium Juri Gagarin, Schwerin
1996	High School Diploma, Douglas High School, Douglas, AZ, USA

Statement of Originality

This is to certify that the intellectual content of this thesis is the product of my own work and that all the assistance received in preparing this thesis and sources have been acknowledged. This thesis has not been submitted for any degree or other purposes.

Eidesstattliche Erklärung

Hiermit versichere ich an Eides statt, dass ich die eingereichte Dissertation selbstständig und ohne fremde Hilfe verfasst, andere als die von mir angegebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Werken wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Diese Arbeit hat noch keiner anderen Fakultät oder Universität zur Prüfung vorgelegen.

.....
Christoph Schmidt

Rostock, November 7, 2024