FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Projector-Based Textures for 3D-Printed Models

Tangible Molecular Visualization

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Visual Computing

eingereicht von

Simon Brenner BSc
Matrikelnummer 0927175

an der Fakultät für Informatik
der Technischen Universität Wien

Betreuung: Prof. Dipl.-Ing. Dr.techn. Ivan Viola
Mitwirkung: Univ.Ass. Dipl.-Ing. Johanna Schmidt
Dr.techn. Manuela Waldner MSc

Wien, 25. September 2015

Simon Brenner | Ivan Viola

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Projector-Based Textures for 3D-Printed Models

Tangible Molecular Visualization

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Visual Computing

by

Simon Brenner BSc
Registration Number 0927175

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Prof. Dipl.-Ing. Dr.techn. Ivan Viola
Assistance: Univ.Ass. Dipl.-Ing. Johanna Schmidt
Dr.techn. Manuela Waldner MSc

Vienna, 25th September, 2015

Simon Brenner

Ivan Viola

Erklärung zur Verfassung der Arbeit

Simon Brenner BSc
Hauptstraße 10
2392 Grub

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 25. September 2015

Simon Brenner

Acknowledgements

I would like to thank Johanna Schmidt for being such a great support throughout the time working on my thesis, regarding matters of both organization and content.

Many thanks to Ivan Viola, who always gave me honest feedback, contributed many new ideas and aspects, and brought me together with renowned researchers, that I could discuss my work with.

I thank Manuela Waldner, who also kept supplying me with valuable comments and information, even though she was on parental leave for most of the time.

Special thanks to Barbora Kozlíková, who contributed a lot to this work. She and her colleagues provided our test data set and produced the high quality 3D-printout, that we used throughout the project. Furthermore, she arranged a meeting with leading researchers at the Loschmidt Laboratories in Brno.

Finally, I want to thank all the people who made time to test and review our system.

This work was supported through grants from the Vienna Science and Technology Fund (WWTF) through project VRG11-010 and by the EC Marie Curie Career Integration Grant through project PCIG13-GA-2013-618680.

Kurzfassung

Durch die zunehmende Verfügbarkeit von neuen 3D-Druck-Technologien sind Molekularbiologen und Molekularbiologinnen nun in der Lage, auf einfache Art und Weise plastische Modelle von großen und komplexen Molekülen herzustellen. Solche Modelle können die mentale Erfassung der zugrundeliegenden räumlichen Strukturen unterstützen. Allerdings sind diese Modelle statisch und meist einfärbig, weshalb ihr Informationsgehalt nicht an bildschirmbasierte Visualisierungsmethoden heranreicht.

Dem Paradigma der *Spatial Augmented Reality* folgend, präsentieren wir eine Methode, um in dynamischer Weise und mit Hilfe eines digitalen Projektors molekulare Eigenschaften direkt auf der Oberfläche von 3D-gedruckten Modellen zu visualisieren. Im Rahmen dieser Arbeit wurde ein Prototyp entwickelt, bestehend aus Hardware und Software, der das Tracking des Modells, sowie das Rendering von farbkodierten Moleküleigenschaften und deren Projektion auf die Oberfläche des plastischen Modells, ermöglicht. Unter Einbezug von Informationen über die Geometrie des 3D-gedruckten Modells, die optischen Eigenschaften des Projektors und die exakte räumliche Beziehung zwischen Projektor und Modell, können die erzeugten Projektionen in Echtzeit aktualisiert werden, sodass sie während einer Benutzeraktion mit dem plastischen Modell registriert bleiben.

Wir untersuchten außerdem die Benutzbarkeit und die potenzielle Anwendbarkeit des entwickelten Systems in Rahmen einer kleineren Benutzerstudie, im Zuge derer wir hilfreiche Rückmeldungen von Experten auf den Gebieten der Biochemie und der Molekularbiologie erhielten.

Abstract

The now widely available 3D-printing technology enables structural molecular biologists to easily produce tangible models of large and complex molecules, which can aid them in understanding their spatial structure. Those models, however, are static and often monochrome, therefore their information content cannot compete with existing screen-based visualization solutions.

Following the paradigm of spatial augmented reality, we present an approach to dynamically visualize molecular properties directly on the surface of 3D-printed tangible models, using a digital projector. We developed a prototype system consisting of hardware and software, that enables the tracking of the tangible model and the rendering of color-coded molecular properties, which are then projected onto the tangible surface. Using knowledge about the geometry of the molecular model, the optical properties of the digital projector and the exact spatial relation between projector and model, the rendered projections are updated in real-time, such that they stay registered with the tangible model during user interaction.

We evaluated the usability and potential applicability of the developed system by collecting feedback from domain experts from the fields of biochemistry and molecular biology.

Contents

CHAPTER 1

Introduction

The project that has been implemented in the course of this master's thesis deals with the exploration of complex proteins using tangible 3D-printed models. To enhance the impression during exploration, a projector-based augmentation is applied to the model, to visualize molecular properties directly on its surface. This chapter describes the motivation for developing this project, gives some insight on possible application areas and describes the methodology used in this thesis.

1.1 Motivation

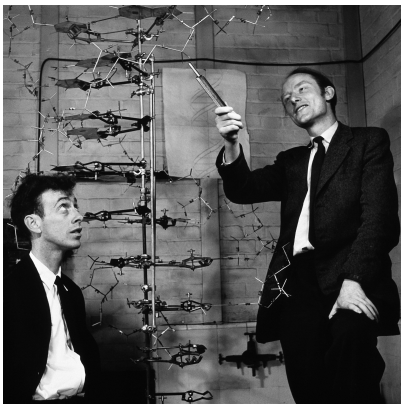
We live in a three-dimensional world, and our senses and perceptional mechanisms have evolutionary developed to perfectly function in this environment. We usually use more than one of our senses when interacting with our environment. When getting to know and learning how to deal with new concepts, we usually try to relate to things we already know. We learn and understand new things best, if more than one of our senses is engaged [HCB12]. Our natural method to learn about the properties of a physical object is to touch it, weigh it in a hand, look at it from different sides or maybe knock on it an hear the sound.

Learning and exploring with our senses, however, does not work all cases. When talking about abstract concepts, like mathematical structures, it is very hard to relate them to objects of the real world. Therefore, our natural perception mechanisms are not directly applicable. In other cases, we would like to learn more about real-world objects, which are too large or too tiny to be touched or seen directly. For example, our solar system can only be explored as a model. On the other hand, molecules need to be represented as enlarged models, because otherwise they would be too tiny to be seen. There are also numerous situations, where objects we would like to explore may only exist in our imagination, yet. For that reason, people have been externalizing their ideas and mental models into physical representations already for a long time. These representations can take the form of paintings and drawings, written words, or three-dimensional models.

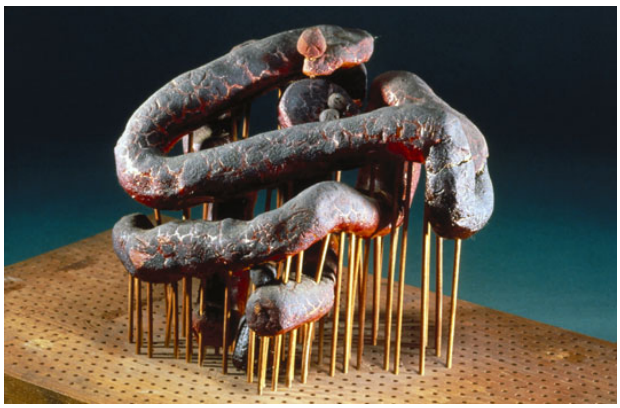
Architects, for example, have a long tradition of crafting miniature models of the buildings they design. This not only helps communicating their ideas in a way that is not possible through mere drawings, but is also a tool for the designers themselves to validate their plans and visions.

Other examples for fields with a long tradition in building models are the fields of chemistry and biology. In both fields people are in many cases dealing with objects of very small size. These objects, nevertheless, exhibit a certain physical structure, that in many cases is important to be understood. For example, all proteins are made of a unique arrangement of amino acids. Such structural information could be drawn on paper, or be described verbally, but both approaches will never fully express the underlying three-dimensional structure.

There are quite a few examples of important discoveries, that were only made through the use of tangible models. Francis Crick and James Watson, for example, discovered that the DNA has a double helix shape by building a model of it (Figure 1.1a). Max F. Perutz and John C. Kendrew built a plasticine model of myoglobin, which is shown in Figure 1.1b, to understand its structure.



(a) Watson (left) and Crick (right) with their DNA model. [his13]



(b) *"Could the search for ultimate truth really have revealed so hideous and visceral-looking an object?"* (Max Perutz) Photo: London Science Museum [lon04]

Figure 1.1: Early models of molecules being used in biology and medicine to understand their physical structure

Although the creation of hand-crafted models helps to get insight into physical structures, it is not always a feasible solution for tangible exploration. Building hand-crafted models takes a lot of time and requires a lot of material. It also requires certain skills and knowledge about how to handle the material, and how to apply these materials to correctly represent the physical structures. Especially in the field of molecular biology, molecules and proteins of interest may exhibit very complicated and complex structures. Even more, such objects may also change over time. However, for many applications, such as inter-molecular docking, the knowledge about the spatial structure of those molecules is of great importance.

In the last decades, desktop computer graphics software became the primary tool for visualizing and exploring spatial molecular structures. The main advantage of the computer-based technologies is their flexibility. It is now possible to freely choose which parts of the molecules are shown on screen, and how they should be represented. There are various ways to visually represent a molecule, which can be chosen depending on the application [KKL+15]. It is, for example, possible to visualize molecules as atom-based ball-and-stick models. In other cases, surface models of the molecule might be preferred. Especially for large proteins, which are basically folded chains of amino acids, the secondary and tertiary structures of those folded chains can be visualized with cartoon-style ribbons, rather than showing the actual atoms. For further analysis, it is possible to visualize properties of the molecule, such as charge or hydrophobicity, directly on the surface. Furthermore, the molecular surface and its inner structure can be shown simultaneously. Figure 1.2 shows an example of the possibilities of computer graphics-based molecular visualization.

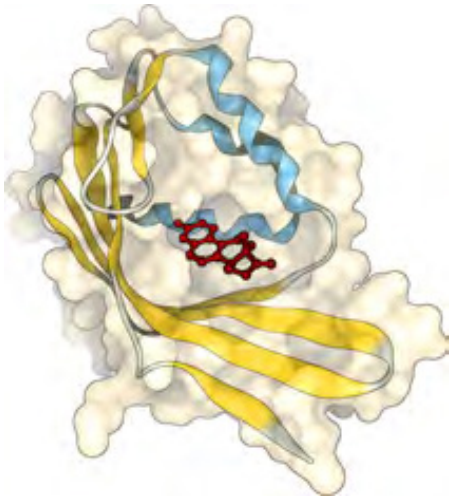


Figure 1.2: A transparent molecular surface reveals the proteins secondary structure, shown as an abstract representation. Additionally, a second molecule (red) is represented as a balls-and-sticks model. [KKL+15]

Despite their flexibility and extensibility, virtual representations of molecular structures also have their drawbacks. First of all, it is always problematic to explore 3D content on a 2D computer screen. Navigation is not as intuitive as when holding an object in one's hands. Usually a series of mouse interactions is necessary for simply rotating the object into a desired pose. It is thus tiresome or hardly possible to gain a mental image of the global shape of such a complex object. An even harder task is the positioning two molecules relative to each other. For the application of inter-molecular docking, which was mentioned before, two molecules have to be positioned precisely such that they touch, but do not overlap. This is an example for a task that could be solved rather easily when dealing with real-world models, but becomes cumbersome when

having to interact with abstract 3D information on a 2D screen.

A recent development in the field of computer science is the research of the usage of 3D-printing for science, industry and manufacturing [Roe11]. With this new technology, it is suddenly possible to transfer virtual 3D-models, that so far only existed as 3D-meshes in the computer memory, into real-world objects. Therefore, this technology became also popular among molecular biologists, because suddenly it was very easy to construct physical molecular models. Objects that could before only be explored on the computer screen, can now be held in one's hands. Such tangible models provide both a source of deeper understanding of spatial structures, and a persistent interface for knowledge transfer between researchers, teachers, and students.

Despite the tangible experience, the 3D printing does not solve all existing problems. The main problem with the 3D-printed models is, that they are static. 3D printers usually print monochrome models. Thus, the printers can translate the spatial structure of the molecule, but they cannot represent additional properties of the molecule (e.g., charge). It would be possible to use very elaborate 3D printers that can print multi-colored objects (see Figure 1.3). However, the properties of a molecule may change over time, and also in many cases multiple properties are of interest. In both cases this would mean that several (expensive) versions of the same molecule have to be printed.

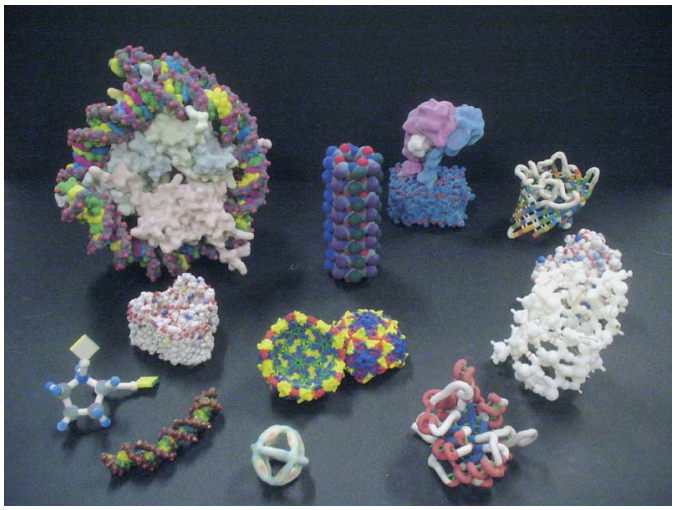


Figure 1.3: Various multi-colored 3D printed molecules [GSSO05]

In this master's thesis we address the problem of having only static 3D-printed molecular models. Our system combines the spatiality and tangibility of the physical models with the flexibility and information content of computer graphics in a single system.

1.2 Methodology

Inspired by the work of Raskar et al. [BR05][RWLB01] on Spatial Augmented Reality, we developed a system that tackles the limitations of having only static 3D-printouts of molecules, which do not show any molecular properties. We use a digital projector to visualize molecular properties directly on the monochrome 3D-printed model. To be able to register the projection with the physical model, the current position and rotation of the physical model, as well as the exact optical properties of the projector must be known. This information can then be used to adapt the rendering of the virtual model to perfectly fit the physical one.

For this purpose we developed a hardware setup, as well as a stand-alone software prototype, which manages the monitoring of hardware devices, the tracking of the tangible object during interaction, and the correctly transformed rendering of the projected visualizations. The system was developed with a focus on affordability and broad applicability, and therefore only uses widely available consumer hardware.

After the implementation of the prototype, we evaluated the applicability and usability of our system by the means of a user study. To this end, we invited domain experts from the field of structural molecular biology to provide feedback on the system. In such a review session, we gave a short demonstration, followed by a hands-on exploration by the domain expert, and a semi-structured interview. The feedback was then used to detect the strengths and weaknesses of our setup compared to other approaches, as well as to identify possible future improvements and extensions of the system.

1.3 Structure of the work

Chapter 2 gives an introduction on molecular visualization in general, explains the use of tangible interfaces and describes the state-of-the-art in tangible molecular visualization. Furthermore, theoretical background and technical approaches, which are important for our implementation and the understanding of the following chapters, are introduced. This includes the paradigm of spatial augmented reality, geometric calibration of cameras and projectors, and object tracking.

Chapter 4 describes the technical details of our prototype implementation. The general hardware setup, as well as the specific devices, that were used to build the prototype, are discussed. The chapter also includes a detailed description of our molecular test data, the calibration approaches we used, as well as a description of geometric considerations and the rendering process.

In Chapter 5, the functionality and the limitations of the prototype are discussed. In particular, we show photos of the system in action and demonstrate its applicability to the field of structural biology. In this chapter we also discuss the limitations of our system, like the working volume and other effects that might influence the exploration experience.

In Chapter 6, we present the results of a user study carried out with domain experts of chemistry and structural molecular biology to evaluate the usability of our system.

Chapter 7 concludes this work by summing up the main insights gained during the project, and suggests future work on improving and extending our approach.

CHAPTER 2

Background and Related Work

The implementation of our proposed projector-based tangible molecular visualization system is a multi-disciplinary project and requires theoretical background from various areas. In the following we will give an overview on fundamental concepts and related work in the fields of molecular visualization, tangible user interface design, spacial augmented reality, calibration and tracking.

2.1 Molecular Visualization

Molecules are by nature three-dimensional. Structural formulas provide information on which atoms are found in a molecule and how they are bonded, but they contain hardly any information about their three-dimensional structure. For certain areas of research, however, this three-dimensional structure is essential; especially when observing the interactions between complex molecules such as proteins. Therefore, other models, which are capable of representing spatial structures, are needed for research and exchange of information.

Traditionally, such models were handcrafted using sticks and modeling clay [B+00]. Nowadays computer graphics and visualizations are dominating the field. In the following a short overview of the main classes of 3D molecular graphic representations which can be derived from modern literature [HDS96] [B+00] [KBE09] [KKL+15] is given:

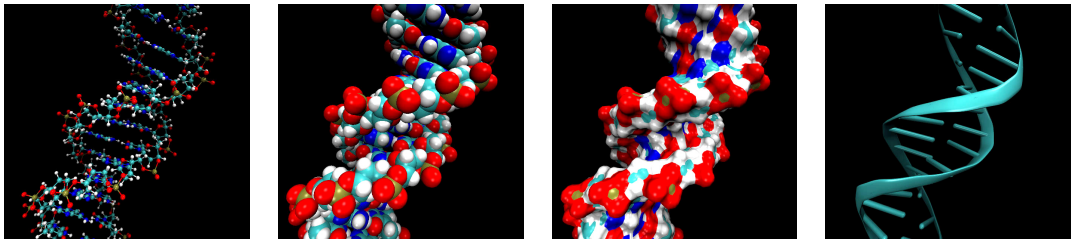
Atom-based models. The balls-and-sticks model was the preferred mode of crafting three-dimensional molecular models before the rise of computer graphics. Bonds are visualized as lines or cylinders (sticks), atoms can be visualized as scaled Van der Waals spheres (balls). This representation is especially useful for representing bonds between atoms. Figure 2.1a shows an example. Space filling models, on the other hand, aim at representing the space occupied by individual atoms. The atoms are represented as spheres, whose radii correspond to their actual sizes, e.g. determined by their Van

der Waals radii. This representation was also initially developed to create plastic models [CP53].

Surface models. The simplest member of this class is the *Van der Waals Surface* (Figure 2.1b), which is based on the space-filling Van der Waals spheres. The surface of the union of those spheres gives the Van der Waals Surface. Extensions are the *Solvent Accessible Surface* (SAS) and the *Solvent Excluded Surface* (SES, Figure 2.1c), which aim at representing the molecular surface that is effectively available for bindings. These representations are the most expressive ones when it comes to interactions between molecules and will be discussed in greater detail in Section 2.1.1.

Cartoons. When molecules are getting too complex, it might be desired to visualize functional structures as simplified abstractions or 'Cartoon' representations, rather than showing every single atom. An example is given in Figure 2.1d.

Volumetric visualizations. Aside from the structure of the molecule itself, domain experts might also be interested in the vicinity of the molecule. Volumetric data like electrostatic potential, density of the surrounding medium, or electrostatic interaction between two molecules can be visualized using conventional volume rendering techniques, such as Iso-Surfacing [LC87] or Volumetric Ray Casting [PPL+99].



(a) Balls and Sticks (b) Van der Waals surface (c) Solvent Excluded Surface (d) Cartoon

Figure 2.1: Different representations of a piece of DNA

2.1.1 Surface representations

The idea of the Solvent Accessible Surface (SAS) and Solvent Excluded Surface (SES) is to represent the surface that is effectively accessible to a given solvent. This is especially interesting for questions concerning docking between different molecules.

The Solvent Accessible Surface, first described by Richards [Ric77], is defined by the center of a sphere as it is 'rolled around' the Van der Waals surface of a molecule. The radius of this sphere is in turn determined by the Van der Waals radius of the specific solvent atoms. In Figure 2.2 the Van der Waals surface is shown in green, the probe is pictured as a yellow circle, and the SAS is shown in blue. One might observe, that the

SAS corresponds to a Van der Waals surface, where the radius of the probe is added to all the Van der Waals radii of the molecule. This 'inflation' can cause gaps, that were present in the Van der Waals surface, to vanish. Obviously, the SAS is not inherent to a molecule, but depends on the used solvent.

The Solvent Excluded Surface (SES) was defined by Greer and Bush [GB78] as the boundary between the solvent and the area where it is 'excluded' by the presence of the molecule. In other words, it is defined by the surface of the union of all possible probes not intersecting the Van der Waals surface of the molecule. The SES is depicted in Figure 2.2 in red. Its advantage over the SAS is that it does not inflate the molecular surface and is thus much better suited for inspecting docking possibilities between molecules, and generally gives a much better picture of the actual molecular surface. It is thus the most widespread molecular surface representation [KBE09]. Just as with the SAS, the surface depends on the assumed solvent.

Due to the characteristics named above, the SES is also an ideal display medium for various surface properties that determine the interaction between the molecule and its surroundings, such as electrostatic charge, hydrophobicity or the ability for hydrogen bonding.

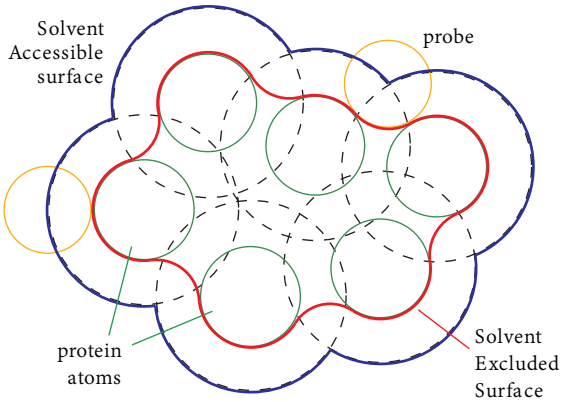


Figure 2.2: Solvent Accessible Surface and Solvent Excluded Surface [KBE09]

2.2 Tangible user interfaces for molecular exploration

The most natural and intuitive way for humans to learn the properties of a physical object is to look at it while touching and manipulating it with their hands. We have used this exploration technique since the first years of our lives and are still doing so in everyday situations. Lederman and Klatzky [LK03] [LK93] did extensive research on haptic object exploration and described a set of exploration procedures that are used to extract certain object properties such as weight, texture, hardness and shape with tactile and proprioceptive senses alone.

Of course, this mode of exploration can only be applied to a limited set of objects. If an object is of abstract nature, not within reach, or simply too big or too small (like a molecule), we have to rely on other exploration techniques. In the case of molecules, researchers typically rely on computer-aided exploration methods based on graphic visualizations as discussed in Section 2.1. It is obvious, that those methods only use the visual information channel.

Furthermore, the user has to deal with the known problems of 3D interaction on a standard workstation [HvDG94]. One of the most important tasks in 3D environments is navigation; or, the dual task in the case of molecular model viewing, the positioning and rotation of an object. Usually the mouse is the primary input device for this task. Position and rotation in 3D space has six degrees of freedom (DOF), three for position and three for rotation. The mouse, however, delivers only two DOF (horizontal and vertical position). This means that the manipulation task must be divided into a series of manipulations, rather than performing a direct manipulation.

The output device, today in most cases an LCD monitor, delivers only a flat projected image of the virtual 3D world. This makes navigation and orientation challenging. Aside from the fact that in real-world environments humans can make use of their stereoscopic vision ability, they also strongly rely on proprioceptive cues as well as the vestibular system (inner ear) for navigation and a sense of position in space. Thus it is more likely to 'get lost' in a virtual world. This also applies to the investigation of a complex 3D model such as a molecule.

The use of tangible interfaces for learning was reviewed by Marshall [Mar07]. He describes possible benefits of tangible interfaces in both exploratory and expressive learning activities, abstract and concrete tangible representations, and various learning domains. The observation of molecules is mentioned as one of the most conclusive applications for tangible interfaces. However, although the benefits for learning are conclusive, intuitive and backed by recognized learning theories, empirical studies on the topic are very sparse. The question if tangible interfaces really lead to better results in learning compared to graphical interfaces, is still an open topic.

2.2.1 Augmented reality with 3D printouts

Gillet et al. [SWS+03] [GSSO05] first proposed to add a tangible component to computer aided molecular exploration. Their work was driven by the thesis, that tangible molecular models are important tools for exploring, understanding and communicating molecular structure. The tactile and proprioceptive senses play an important role in understanding and manipulating 3D shapes; in conventional visualization approaches, those senses are omitted. In the past, it was common to work with tangible models. Traditionally, atom-based models were fabricated out of sticks and modeling clay, wire or other materials; but especially for complex proteins, those models are on the one hand not feasible to construct, and on the other hand inappropriate for many applications like inter-molecular docking.

Inspired by the increasing availability of 3D printing technology, Gillet et al. used printed SES (see Section 2.1.1) models as a basis for their tangible interface approach.

Those models are by nature rigid, static, and in most cases monochrome. To combine the tangible models with the flexibility and information content of conventional visualization software, the researchers developed an Augmented Reality (AR) application, which will be briefly described in the following.

The printed model is captured by a camera connected to a PC. There are markers attached to the model, which allow for optical tracking using the ARToolkit [KB99] (see Section 2.5.1). Once the position and orientation of the model in relation to the camera is known, virtual content can be rendered 'on top' of the camera frame, registered with the image of the physical 3D printout. The composite image is then displayed on a monitor. A user can now hold the molecular model in his hands, touch it and move it around freely, while watching any additional visualizations on a screen. Figure 2.3 shows a user inspecting a Ribosome using the described setup. The small subunit of the Ribosome is printed, the large subunit is rendered as a wireframe model. Additionally, the position of the tRNA passing through the Ribosome is visualized in red.

(a) Setup (b) Screenshot

Figure 2.3: A user exploring a Ribosome [GSSO05]

It is worth noting that in this implementation, the tangible model and the additional visualizations are spatially separated; actually, it can be seen as a tangible six DOF input device for conventional screen-based molecular visualization, that enables direct manipulation and a tactile impression of the molecule. The authors already mentioned the possibility to use a see-through head-mounted display instead of a monitor, which would eliminate the spacial separation of contents and make the AR experience more immersive; or, as another option, to place the visualizations directly on the printout using a projector. However, the authors found that at the time of writing, those technique were too expensive and immature to be feasible for a broad application; that is why they sticked to the desktop monitor solution.

User studies have been performed in a teaching environment as well as in a research context. The authors claim that they received uniformly positive results. In the teaching area the test subjects found the new interface engaging and instructive; researchers saw potential in a more efficient communication and comprehension of structural properties of molecules.

To summarize the advantages of the presented tangible interface over conventional desktop solutions:

- The interaction is a multisensory experience which supports learning
- Direct manipulation of the virtual content (not through a series of mouse operations). This is mainly an ergonomic aspect.
- Natural overview and detail, when viewing the model from various distances.
- Persistent objects are produced; they can be shared between collaborating individuals and may enhance knowledge sharing and social interaction.

2.2.2 Other approaches

There has also been research on a fundamentally different approach to haptic molecular exploration [SB08][CM10]. Rather than creating a tangible model of a molecule, the forces acting between molecules are calculated in real-time and displayed using a haptic rendering device. This is useful for finding docking sites between two molecules. The user can feel how the molecules are interacting and thus solve the docking site problem like a physical puzzle.

Schkolne et al. [SIS04] on the other hand, proposed an immersive virtual environment to design DNA molecules. In their setup, the user wears shutter glasses while working on a responsive workbench, thus perceiving the molecule to be floating above the workbench surface. Specially designed input devices are used to manipulate the molecule. In this case, not the molecules themselves are 'tangible', but the input devices used to manipulate them.

2.3 Spatial Augmented Reality

The main idea of our approach, the texturing of real world objects using a projector, is based on the work of Bimber and Raskar [BR05] on *Spatial Augmented Reality* (SAR). The main distinction of SAR with respect to traditional Augmented Reality is the media on which artificial content is displayed. Whereas Augmented Reality in general relies on head-mounted displays (HMDs) or hand-held devices, SAR tries to find ways to inject content directly into the real world. This can be achieved by transparent screens, beam combiners, hologram techniques or projections, just to name a few.

We will discuss only the latter technique in detail, as it is the basis for our work. In 2001, Raskar et al. [RWLB01] introduced the concept of *Shader Lamps*. The approach is based on the insight, that the visual appearance of an object solely depends on the wavelengths of light that enter our eyes from the direction of the object. In our everyday surroundings, these wavelengths are typically produced by a source of uniformly colored light (e.g. sunlight), which hits, possibly after bouncing off other surfaces, the object of interest. Parts of the incident light are absorbed by the object, other parts are reflected;

only the reflected parts reach our eyes and create the sensation of a specific color and intensity. Mathematically this can be described by the rendering equation [Kaj86]

$$I(x, x') = g(x, x') \cdot \left[\epsilon(x, x') + \int_S \rho(x, x', x'') \cdot I(x', x'') dx'' \right]$$

which describes the amount of light that arrives at point x from point x'. In this expression $g(x, x')$ is a 'geometric term' (incorporating occlusion, distance, etc.) and $\epsilon(x, x')$ is the light *emitted* from x' to x. The integral accounts for light arriving at x' from all points x'' visible from x'; multiplied by the *Bidirectional Reflectance Distribution Function* (BRDF) $\rho(x, x', x'')$, which gives the amount of light from direction of x'' that is reflected in direction of x at a surface point x'.

The basic idea is, that a colored object illuminated with neutral light could be replaced with a neutral object illuminated with colored light, producing the same image on the retinas of a spectator. In mathematical terms: for a different $\rho(x, x', x'')$ (another material) we can calculate the light necessary from the direction of the projector $I(x', x^{projector})$ to produce a fixed intensity $I(x, x')$ in the user's eye.

For example, a block of wood could be replaced by an equally shaped white cardboard box, augmented with a corresponding texture projection. Theoretically, such a projection could make the augmented cardboard box visually undistiguishable from the wood block. Practically, due to insufficient physical models, lack of information about the environment or simply the limited color reproduction of projectors, this is nearly impossible to achieve.

However, *Shader Lamps* can do more than trying to produce realistic replicas of real objects. This technique allows to change the visual properties of objects in real-time and on demand. This can be useful for many applications: simulating different lighting situations on architectural models, interactively painting on the model, as proposed in a successive work [RBF01], quickly changing theater backgrounds, large-scale projections on buildings, and much more.

There are some differences to traditional rendering that have to be taken into account when attempting projecector-based texturing of real-world objects. In the following, the fundamental techniques will be discussed.

2.3.1 Projection onto non-planar surfaces

Typically a projector is used to project an image onto a planar display surface, thus creating a rectangular image, just like an image displayed on a CRT or LCD monitor. However, in contrast to monitors, projectors can be used to create non-planar images by projecting onto non-planar display surfaces.

If a user is located at position T and should perceive a point of a virtual object in space at the position V, then this point must be projected onto the display surface at point M, where the line (T, V) intersects the surface. Then, under knowledge of the intrinsic and extrinsic parameters of the projector (see Section 2.4), the point m_p in

projector image space, which is illuminating M, can be calculated. This is illustrated in Figure 2.4.

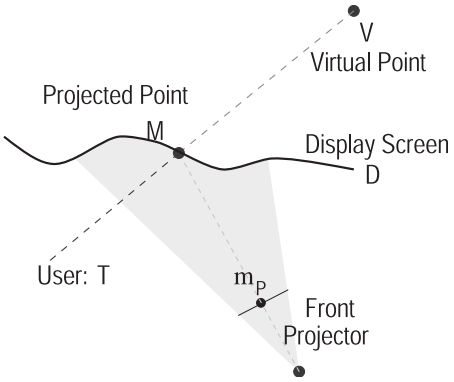


Figure 2.4: Projection onto a non-planar surface [BR05]

One straight-forward way to implement such a rendering system is the following two-step algorithm (see Figure 2.5):

1. Render the virtual scene from the user's point of view and store it in a texture.
2. Render the display surface from the projectors point of view, while virtually projecting the texture from step one onto the surface from the user's point of view.

The projection of the texture onto the display surface in step 2 can simply be accomplished by projective texture mapping [Eve01], a technique that is mostly used in the context of shadow maps [Wil78]. The resulting image can then be physically projected onto the display surface.

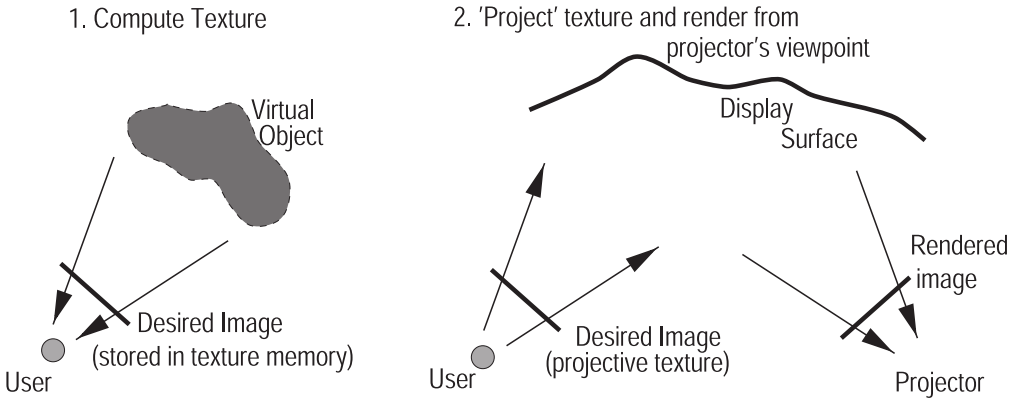


Figure 2.5: Two-step rendering algorithm for non-planar surfaces [BR05]

For a correct rendering, the following information has to be taken into account:

- the position of the user,
- the exact geometry of the display surface,
- the position and orientation (extrinsic parameters) of the projector, and
- the intrinsic parameters of the projector.

A special case occurs, however, if the surface of the virtual object coincides with the display surface. This is the case in classical *Shader Lamps* applications, which aim at altering surface properties of physical objects. This means we want to render a virtual object that is geometrically equivalent to, and perfectly registered with, the physical object (=the display surface).

In this case, the rendering process is drastically simplified: The first rendering step can be skipped entirely. The virtual object only has to be rendered from the projectors point of view and projected on the physical object. Also the position of the user must not necessarily be known, if no view-dependent shading effects (like specular highlights) are desired.

2.3.2 Intensity correction

According to Lambert's Cosine Law [PPP06], the luminous intensity of a diffuse surface depends on the angle between the surface normal and the direction of incident light. This also applies to a projector illuminating a non-planar object: when illuminating different parts of the display surface with the same intensity, the steeper parts with respect to the projection direction will appear darker. To compensate for this effect, Raskar et al. propose an intensity correction that can be seen as an 'inverse Lambert Shading'. The intensity of an output pixel is weighted by $1/cos(\alpha)$, where α is the angle between the surface normal and the direction from the surface point to the projector. As this expression approaches infinity as α approaches 90°, this model is not really applicable as such and requires a couple of heuristics and adaptations. For example, Raskar et al. simply did not illuminate surfaces with an α greater than 60°.

Additionally, one might want to use a setup of multiple projectors from different directions, to achieve a more complete illumination of the object. This leads to another intensity-related problem, namely how the images of different projectors can be combined to create one smooth and seamless projection. In multi-projector setups using planar projection surfaces, the problem is solved by using various feathering techniques that are also used in image processing for stitching mosaique images [UES01]. In overlapping regions the contribution of each source image is then weighted by its alpha function, which depends on the distance to the closest border of the image. Obviously, this approach can also be used for non-planar surfaces. To determine the overlapping regions, an algorithm similar to shadow-mapping can be used, where the projectors take the place of light sources.

Problems occur on concave surfaces, where some areas in overlapping regions can not be illuminated by all of the overlapping projectors. This leads to discontinuities in the blending function, as illustrated in the upper plot of Figure 2.6. If the projectors have different color gamuts or luminances, or the geometric calibration is imperfect, it will lead to visible edges in the projected image. The discontinuities at c and e could be avoided by adapting the alpha functions, such that it depends not solely on the distance to the nearest border, but also on the distance to the nearest discontinuity (Figure 2.6, bottom plot). Raskar et al. propose an algorithm to compute such an alpha function [RWLB01], but due to its computional complexity, it is only feasible for static setups, where it is not necessary to recompute the blending functions every frame [RBF01]. Discontinuities as seen at d cannot be avoided in any case.

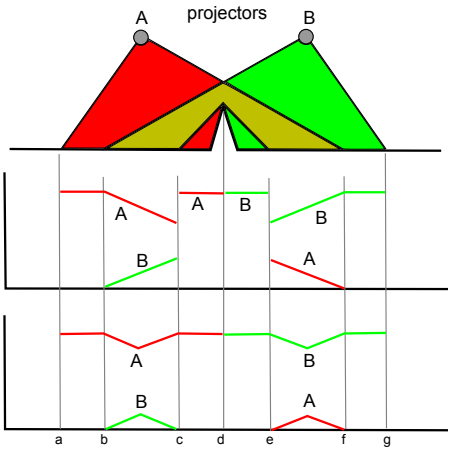


Figure 2.6: Projection overlaps on concave surfaces. Image modified from [RWLB01].

A completely different approach especially designed for projector blending on movable non-planar surfaces was proposed by Lincoln et al. [LWF11]. Here the alpha function is not based on the distance to image-borders, but on the directions of surface normals with respect to the individual projectors; which works in real-time.

In the context of multi-projector blending one could also ask how many projectors are necessary to illuminate every point of a surface, and how they should be positioned. This generally NP-hard problem is related to the *Art Gallery Problem* [O'R87] (how many cameras are needed to observe every point in every room of an art gallery).

2.3.3 Issues

There are some inherent properties of projections that limit the capabilities of *Shader Lamps* and should be kept in mind:

- **Depth of field.** Just as cameras, projectors have a limited depth of field, that is, projections are only sharp in a certain distance range from the projector.

- **Secondary scattering.** Depending on the display surface, secondary scattering between neighboring surfaces can hardly be avoided. Thus, it is difficult to simulate very matte surfaces. On the other hand, in certain applications the scattering might be desired and comes for free.
- **Shadows cast by users.** Naturally, when a user is interacting with the augmented object, his/her hands will block some of the projector's light.
- **Ambient light.** In most usage scenarios, there will be light sources other than the projectors that will have an impact on the appearance of the augmented object. Especially white light will wash out the colors.

2.3.4 Applications of projector-based SAR

In the following we will describe other projector-based spatial augmented reality applications, to give an overview on the state of the art in that area.

The *reacTable* [JGAK07] is a tangible, visually responsive musical instrument. It is based on a translucent table, with a camera and a projector are placed underneath. Users can place and manipulate building blocks on the table, which are tagged with fiducial markers. The camera tracks those markers; their position and rotation serve as controls for the synthesizer. The projector is used to visualize the music and the state of the system on the translucent surface (Figure 2.7a).

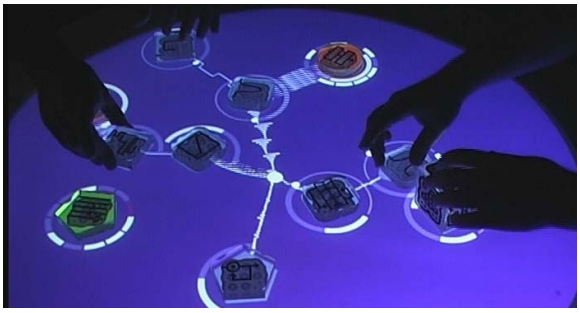
Dalsgaard and Halskov [DH12] describe a system similar to the *reacTable*, but extended to be used for general visualization purposes. In addition to the back-projection from underneath the translucent table, there are projectors mounted above the table which project onto tangible building blocks. Those tangible elements, thus, become both input devices and three dimensional display surfaces. Figure 2.7b shows the tangible cubes with statistical data projected onto them, depending on their position on a map.

RoomAlive by Microsoft Research [JSM+14] turns a whole room into an immersive virtual environment using projector-camera units. The modular system is made of multiple units of a projector and a Microsoft Kinect depth camera. The system creates a 3D model of the entire room and can then create an arbitrary virtual environment. The user's body motions are tracked, enabling for example immersive gaming experiences. Figure 2.7c shows an example of a virtually textured room.

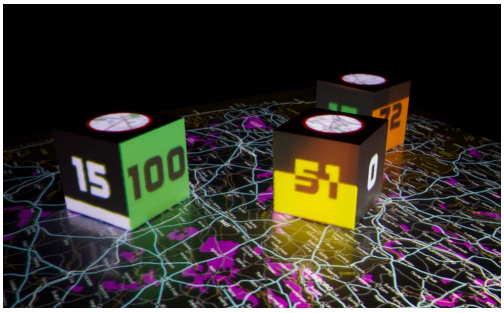
Parker et al. [PLS+15] proposed the use of *Shader Lamps* in product design evaluation. A neutrally colored mockup model of a product (such as a car) can be augmented with projections to preview the appearance of the model if made of a certain material, or painted with a certain color. This helps to save time and money in appearance design studies, as one mockup can be re-used for numerous material proposals. To achieve the required realism, the authors propose a framework including a measurement-based material database for both BRDF and subsurface scattering material models, an according high quality rendering, and a color normalization to account for different projector gamuts and materials of the projection surface. An example can be seen in Figure 2.7d.

Teegi (Tangible EEG Interface) [FGF+14] is an SAR application that visualizes a user's own brain activity on a tangible avatar. The avatar, as well as further objects that

serve as tangible input devices to change the properties of the visualization, are tracked optically and can be placed at arbitrary positions on the table. This system was designed to make electroencephalography (EEG) technology more accessible and understandable to the general public. An example can be seen in Figure 2.7e.

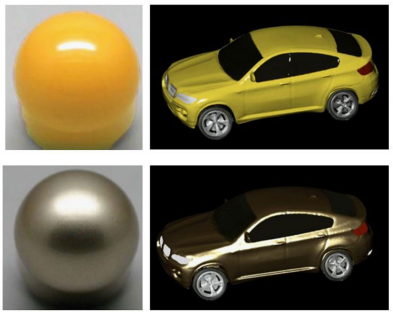


(a) The *reacTable* operated by multiple users [JGAK07]



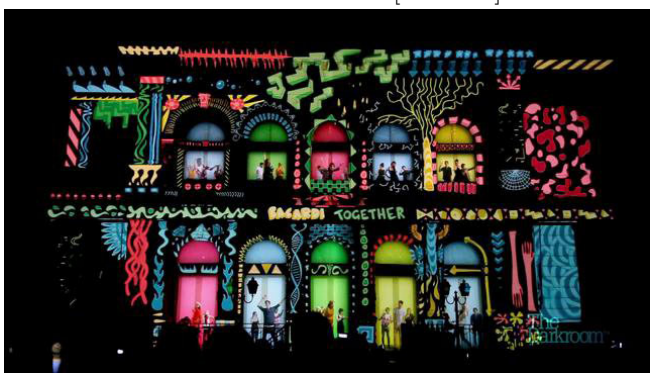
(b) *Tangible 3D tabletops*: 3D visualization [DH12]

(c) *RoomAlive*: a room augmented with artificial textures [JSM+14]



(d) SAR for product design evaluation [PLS+15]

(e) Teegi: Tangible EEG interface [FGF+14]



(f) Projection mapping for advertisement on the *Kursalon*, Vienna [kur11]

Figure 2.7: Examples for projector-based SAR

Within the last years, projector-based SAR became increasingly popular in arts, entertainment, and advertising under the synonym 'projection mapping'. Especially

large-scale projections onto architectural objects draw a lot of attention. Figure 2.7f shows an example of such a projection show on the *Kursalon*, Vienna, for advertisement purposes. There have also been proposals to apply this technique to cultural heritage preservation [FDS12][Cat13]. Previous building attributes or even construction plans can be projected onto the building in its present-day state, thus creating an AR time travel for the audience.

2.4 Calibration

In the following, the term *calibration* will exclusively refer to the geometric calibration of cameras and projectors. Color calibration will not be discussed here.

The goal of camera calibration is to determine the extrinsic and intrinsic parameters of a given physical camera. The extrinsic camera parameters describe the position and orientation of the camera with respect to a global coordinate frame. The intrinsic parameters refer to the inner structure of the camera. Once they are known, a camera is basically upgraded to a directional light sensor: for each pixel, the direction from where it is illuminated can be computed. This property is essential to any computer vision application where a camera is used to perform exact geometrical measurements of the real world, and for augmented reality applications like the one proposed in this work.

The *pinhole camera model*, which will be briefly discussed in Section 2.4.1, is the predominant model used to describe a camera. Aberrations from this model, such as lens distortion (because most cameras use lenses instead of pinholes) can be described by an additional distortion function.

Projectors can be seen as inverse cameras. Their behavior can also be described by a camera model, only the flow of information is reversed. Thus, we will describe the basic models and calibration approaches only for cameras, implying that they can equally be applied to projectors.

After discussing the underlying camera model, we will introduce some specific algorithms to calibrate cameras, projectors and combined systems in Section 2.4.2.

2.4.1 The pinhole camera model

The geometric properties of most cameras can be roughly described by the pinhole camera model; its parameters are, aside from additional lens distortion coefficients, the basic result of a camera calibration. Therefore, as a theoretical background for calibration, the following paragraphs will provide a short overview on the model.

A pinhole camera operates on the same principle as the historical *Camera Obscura*. It can be pictured as a closed box with a little hole in one of its walls. This makes sure that each point on the opposite wall (where in the case of a camera a film or sensor is placed) is only hit by light rays from outside the box coming from a single direction. Thus, a sharp mirrored 2D image of the 3D outside world is projected onto this surface. Such a camera has an infinite depth of field and no lens distortion, but the luminous efficiency is very low.

This simple optical device is the prototype for the pinhole camera model. Figure 2.8 shows the main components. The *camera center* **C** is the center of projection. The *image plane* is located at a distance of focal length f from the camera center. In contrary to a real pinhole camera, where the projection target is behind the center of projection and the resulting image thus mirrored, it is more convenient to assume the image plane to be in front of the camera center. The *principal axis* is the line through the camera center perpendicular to the image plane; it intersects the image plane at the *principal point* **p**.

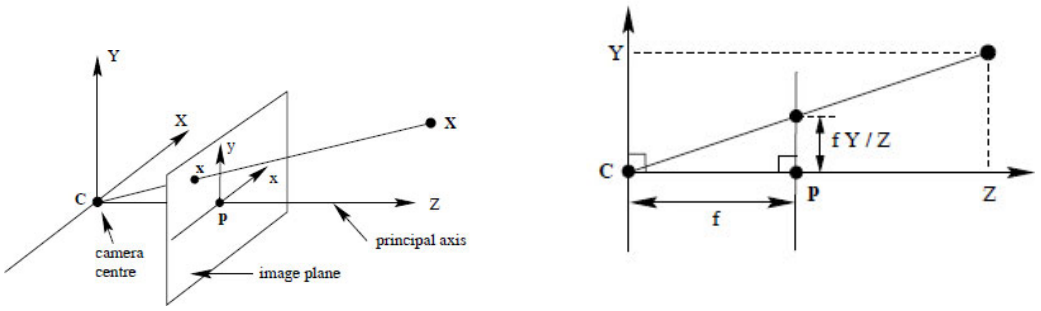


Figure 2.8: The pinhole camera model [HZ03]

The pinhole camera model now maps any point in space **X** to the intersection of the line joining **C** and **X** with the image plane. This perspective projection is a linear mapping of a 3D point in world space **X** to a 2D point in image space **x** and can be expressed as a matrix multiplication when using homogeneous coordinates:

$$\mathbf{X} \mapsto \mathbf{x} = \mathbf{P}\mathbf{X}$$

The projection matrix **P** is a 3×4 matrix, which can be decomposed as follows:

$$\mathbf{P} = \mathbf{K}[\mathbf{R}|\mathbf{t}]$$

where **R** is a 3×3 rotation matrix and **t** is a translation vector. So the block $[\mathbf{R}|\mathbf{t}]$ contains the *extrinsic parameters* of the camera, that is, the position and rotation of the camera center in relation to a global coordinate system. The more interesting part is the 3×3 matrix **K**, the 'intrinsic matrix' or 'calibration matrix'. This matrix incorporates the *intrinsic parameters* of the camera.

$$\mathbf{K} = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix}$$

α_x and α_y correspond to the focal length. They are different for x and y direction if the pixels are non-square. x_0 and y_0 define the offset between the origin of image space and the principal point in pixels. s is an additional parameter to account for pixel

skewness, which is zero in most cases. **K** has 5 degrees of freedom. For a detailed and comprehensive derivation of the above matrix the interested reader is referred to the book by Hartley and Zissermann [HZ03].

As mentioned above, real cameras deviate from the pinhole model in that they use lenses for focusing. This leads to a better luminous efficiency, but introduces distortions and a limited depth of field. *Radial distortion* is the most relevant aberration; in general, it increases with the field of view of the lens, but also the quality of the lens has an impact on distortion. The radial distortion can be modeled with a function defining a deviation of points in image space from the coordinates that would result from a projection following the ideal pinhole model. This function is non-linear and depends on the distance from the principal point in image space. Additionally the *tangential distortion* can be modeled. This kind of distortion occurs if the camera sensor and the lens are not aligned in parallel. Compared to the radial distortion, those distortions are very low in conventional cameras.

2.4.2 Calibration algorithms

Calibration aims at determining the intrinsic and extrinsic camera parameters. The intrinsic parameters are inherent to a specific camera. The extrinsic parameters may change over time, and depend on the chosen global coordinate system. They are especially important in systems involving multiple cameras, camera movement over time, or camera-projector setups.

A common principle of camera calibration algorithms is the finding of point correspondences in image space and world space; a set of pictures taken of a scene by the camera is mostly a fundamental part of the input. The algorithms can be classified depending on what information is known about the captured scene [Zha04]. On one side of the spectrum there are methods using well-defined three-dimensional calibration objects, whose geometry is known to a high precision. For example, two or three orthogonal planes, or a plane undergoing well-defined translations between the frames. Those methods are very precise and efficient, but require an expensive setup.

On the other side of the spectrum, we find the auto-calibration methods which do not use special calibration objects. Constraints on the camera parameters are given by point correspondences of different views on the same rigid scene. Obviously, the advantage is that no additional equipment is needed. However, those methods are mathematically more complex, as a larger number of parameters has to be estimated, and thus more error-prone.

The calibration method by Zhang [Zha00] can be seen as a hybrid and is certainly the most widespread one, because of its ease of use and good performance. There exist numerous free-to-use implementations of the algorithm [Bou08][Bra00]. It uses a calibration object, but a rather simple one that can be easily produced: a plane with a printed calibration pattern attached. In principle, any pattern can be used, as long as a set of points can be uniquely detected. In practice, just as proposed by Zhang himself, a checkerboard pattern is used. The algorithm can be outlined as follows:

1. Take pictures of the calibration plane in different positions and orientations
2. Detect the defined points of the calibration pattern on each image
3. For each image, estimate a homography between the pattern plane and the image plane using the point correspondences
4. Each homography gives two constraints on the intrinsic parameters. As the intrinsic matrix $\mathbf{K}$ has 5 DOF, we need at least 3 homographies/input images to solve $\mathbf{K}$.
5. Compute the extrinsic parameters $\mathbf{R}_i$ and $\mathbf{t}_i$ for each pattern plane i
6. (optional) Compute an initial guess for the distortion coefficients
7. Maximum likelihood estimation of all parameters by minimizing the re-projection error

For more details and mathematical background the reader is referred to [Zha00][Zha04][HZ03].

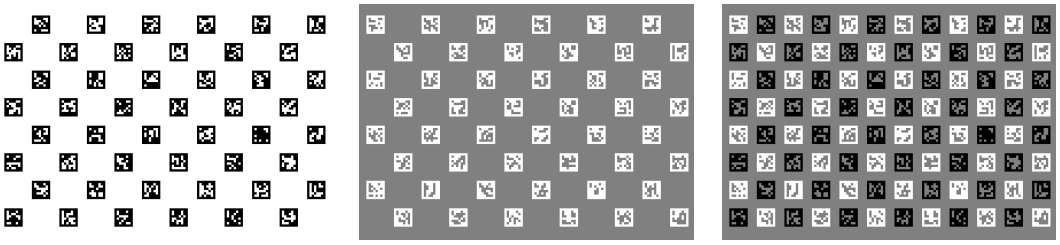
Although projectors can be seen as inverse cameras and described by the same model, their calibration holds additional challenges. When calibrating a camera, points of a known object in 3D space must be found in the camera image. When calibrating a projector, points of a known 2D image, that is projected into the scene, must be found in 3D space; this is much harder. Also, in contrast to cameras, projectors have no possibility to observe their environment. So normally an additional camera, which is in a static relation to the projector, has to be used.

There also exist methods that work without additional cameras. Raskar et al. [RBF01] calibrated the projectors for their Shader Lamps setup by projecting crosshairs into the scene and measuring their positions with a tracked stylus, thus creating point correspondences between projector image space and 3D space. This procedure is both expensive and tedious.

Another method [FHMF09], that uses a camera, is based on a camera calibration following Zhangs method. In addition to the printed pattern, another pattern is projected onto the calibration plane by the projector while the input images are taken. First the camera is calibrated using the printed patterns. After that, the external parameters, that is, the transformation of the camera center with respect to the pattern, are known for each input image. Thus, it is possible to determine the equation of each calibration plane in the coordinate system of the camera. Next, the points of the projected patterns are detected on each image and, as the camera is already calibrated, can be converted to rays through the camera center and the image plane. The intersection of those rays with the previously computed calibration planes gives the 3D positions of the projected pattern. Those positions can then be used to setup 2D-3D-correspondences and compute the projector calibration.

Audet and Okutomi [AO09] proposed the so-far most user-friendly method for camera-projector calibration, based on fiducial markers. For this method no pre-calibrated camera is necessary; camera and projector are calibrated at the same time. As a calibration

pattern, not a checkerboard is used, but an array of ARToolkitPlus markers (more details can be found in Section 2.5.1). One such array is printed and attached to a planar surface. The other array is projected on top of the printed one in an interlaced manner; Figure 2.9 gives an example of the printed and projected pattern.



(a) Printed pattern (b) Projected pattern (c) Perfect alignment

Figure 2.9: Calibration patterns used by Audet and Okutomi [AO09]

Based on markers of the printed and projected patterns detected in the camera image, a homography is estimated, which is used to pre-warp the projector image such that it is perfectly aligned with the printed pattern (see Figure 2.9c). This is usually not achieved in the first try, but is iterated until a sufficiently good alignment is reached. The resulting configuration is then saved for calibration, the user is required to change the pose of the printed pattern, and the process is repeated. After a couple of such configurations (Audet and Okutomi recommend at least ten), the calibration of the camera-projector system (intrinsics of camera and projector, and the transformation between camera center and projector center) can be computed. The precision is on a level with other, less user-friendly methods.

2.5 Object tracking

In various applications, including augmented reality, the position and rotation of real-world objects at a given point in time is of interest. The acquisition of this information is called tracking. If both the position and rotation of an object are determined, it is often referred to as six degrees of freedom (DOF) tracking (three DOF for the position in 3D and three for the rotation). Tracking can be accomplished using various technologies [WF02]:

Mechanical: The tracked object is attached to a mechanic apparatus with joints, that allow the movement of the object. Typically the current pose is calculated from angular sensors in the joints. This tracking modality can be extremely fast and precise; on the downside the movement of the object is strongly restricted and the devices are expensive.

Inertial: Inertial sensors measure linear or angular accelerations. The position and rotation can be computed by integrating the accelerations. The main drawback of inertial

tracking is the so-called *drift*: Over time errors accumulate and the estimated pose 'drifts' off. Therefore this tracking modality should be used in combination with another modality, that can 'reset' the pose in certain intervals.

Radio: The object of interest is equipped with a radio transponder. Receivers are placed at known positions. If the transponder sends a signal, its position can be triangulated from the different arriving times at the receivers. This is the same principle as that of the Global Positioning System (GPS), just with transponders and receivers interchanged. To achieve six DOF, three transponders must be placed on the object. The precision of radio tracking systems is usually low, but they are well-suited for large tracking volumes (e.g. football fields).

Ultrasound: The principle is the same as in radio tracking, but ultrasound waves are used instead of radio waves. Sound travels much slower than electomagnetic waves, which leads to an increased precision compared to radio tracking. However, the working volume is much smaller, air-pressure and temperature alter the speed of sound, and objects between sender and receiver may disturb the tracking.

Magnetic: A transmitter generates three orthogonal magnetic fields. The receiver attached to the object is equipped with three orthogonal coils to measure those magnetic fields. Those trackers have a high precision and typically a very restricted working volume. The main disadvantage is their sensitivity to magnetic distortions of the environment. On the other hand, like radio tracking, a direct line of sight between sender and receiver is not required.

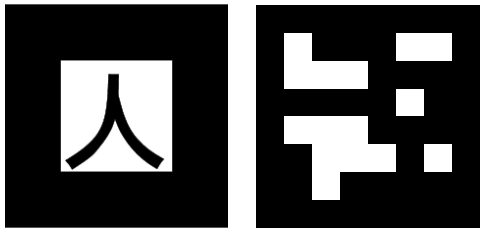
Optical: The pose of an object is retrieved from video frames captured by one or more cameras. As there are numerous fundamentally different approaches and optical tracking is of special interest for this work, it will be discussed in further detail in the following subsection.

2.5.1 Optical tracking

A major distinction can be made between marker-based and marker-less tracking. In approaches of the former class, a well-known marker, which is attached to the object, is tracked instead of the object itself.

One old but still frequently used marker-based tracking framework is the **ARToolKit** [KB99]. The markers can be easily produced as they are two dimensional printed patterns. The markers consist of a quadratic border and a symbol that ensures the unique identification and rotary unambiguousness of the marker (Figure 2.10a). As long as the size of the marker and the calibration of the camera are known, the position and rotation of the marker in space can be derived from the distortion and rotation of its image in a single camera frame. The ARToolKit was designed for augmented reality applications and, because of its cheap and easy usage and its good performance, has been used in many projects [MK12]. A popular extension of the ARToolKit is the software

library **ARToolKitPlus** [WS07]. Instead of arbitrary images, it uses BCH-Id markers (see Figure 2.10b).



(a) (b)

Figure 2.10: Marker of ARToolKit and ARToolKitPlus

Another class of marker based optical tracking systems, such as the one implemented by Pintaric and Kaufmann [PK07], uses three dimensional markers in form of sets of uniquely arranged retro-reflective balls. The markers are viewed by at least two synchronized, calibrated infra-red cameras. Those cameras are equipped with infra-red strobes; the emitted light is reflected back to the camera by the retro-reflective balls, making them clearly visible in the camera image as bright spots. Using stereo vision techniques, the pose of the marker can be extracted from the synchronized images.

There also approaches using active markers, which emit light instead of only reflecting it. Those approaches typically rely on infra-red LEDs [MWP06][FMSS14]. Mossel et al. [MGV+14] applied a tracking framework using active infra-red markers to automated tunnel construction.

Marker-less optical tracking, or natural feature tracking, operates on the tracked objects themselves. A prominent example is the human pose tracking used by the Microsoft Kinect [SFC+11]. An infra-red point pattern is projected into the scene. The scene is then captured by an infra-red camera and depth information is extracted from the distortions in the pattern. Through excessive machine learning, the algorithm is trained to extract body parts and poses from the depth image.

For the marker-less tracking of rigid objects of known geometry, most approaches are based on the detection of edges [BB07], texture features, or both [LSFK10][PMK13].

2.5.2 Filtering

In general, sensor data contain noise to some degree, introduced by inaccuracies of the sensor, or errors during processing. Tracking devices are no exception, and those inaccuracies can result in noticeable jitter. The common way to get rid of sensor noise is the application of some kind of filter. A trivial solution would be to compute a (weighted) average of the sensor data of the last x frames to smooth out the noise. However, this is suboptimal, because past frames contributing to an average lead to a delay.

A desirable solution would estimate the current state of the system, under consideration of past states, in an optimal way. The *Kalman filter* [Kal60], which is based on

Bayesian statistics, was developed for this purpose. It consists of the alternating steps of prediction and correction. During the prediction step, the current system state is estimated based on a probability distribution. During the correction step, a new sensor data sample that comes with an uncertainty itself is used to update the probability distribution. Depending on the deviation between the predicted and the measured state, the predicted state ore the measured state is given more credability (the *Kalman Gain*).

Rubio et al. [RQPR$^+$06] proposed the use of a Kalman filter for 3D optical tracking applications to eliminate jitter.

CHAPTER 3

Proposed Approach

The goal of this master's thesis was to implement a system, consisting of software and hardware, which enables the augmentation of 3D-printed molecular models with registered digital projections. The application incorporates tracking, the visualization of 3D molecular models and their correct projection on the tangible model, and provides a graphical user interface that enables the user to modify the projection output and monitor the involved devices.

One major goal was to design the system in a way that is affordable and can be assembled using mainly widely available components. An overview is shown in Figure 3.1. The system consists of:

A 3D-printed molecular model. It serves as a tangible interface and a canvas for visualizing molecular properties.

A digital projector. This is the core tool to visualize patterns directly on the tangible molecular model.

A digital camera. This camera is fixed to the projector and is mainly used to calibrate the projector.

An infra-red camera. Another digital camera, which operates in the infra-red spectrum and is used for optical tracking.

An infra-red light source (optional). If the lighting environment does not provide sufficient amounts of infra-red light, or if the visual appeal of the projection should be improved by dimming visible light, the use of an additional infra-red light source is recommended.

A personal computer. It serves as the main workstation and is used to run the software, coordinate the other devices and produce the graphics output.

In the following Sections, the proposed system will be discussed in greater detail.

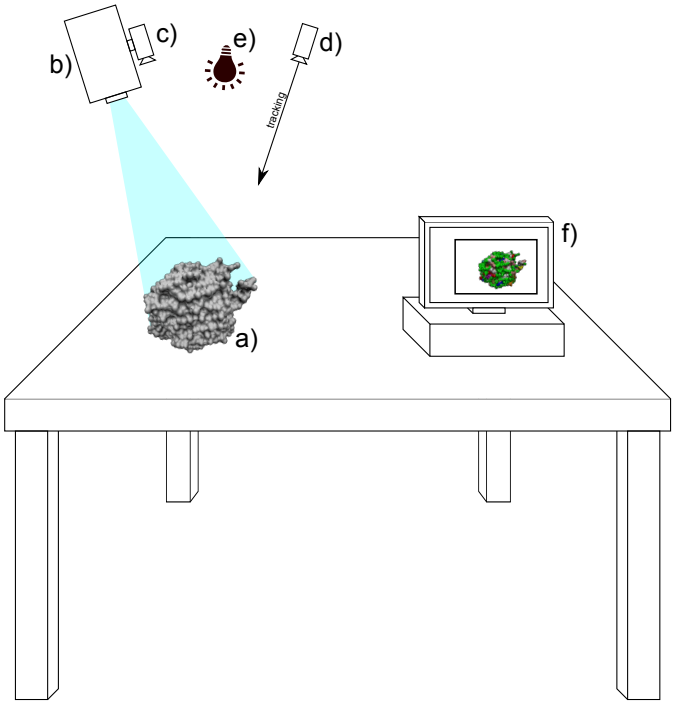


Figure 3.1: An overview of the hardware setup: a) the 3D-printed molecular model b) digital projector c) digital camera d) infra-red camera for tracking e) optional infra-red light source f) personal computer

3.1 The tangible model

Our system is specialized on solvent excluded surface (SES) models of proteins. We opted for the SES representation, because it is considered the most informative surface representation. Furthermore, its smoothness suits our setup well, because small inaccuracies of the projection are not as obvious as when using models with distinct edges, such as a Van der Waals surface.

The model should be printed in a uniform and neutral color, such as white and gray, so that the projected colors are not distorted. Furthermore, it should be printed in a size that makes it convenient to handle and still large enough to exhibit the molecule's structural features. Figure 3.2 shows such a suitable 3D-printed molecular model.

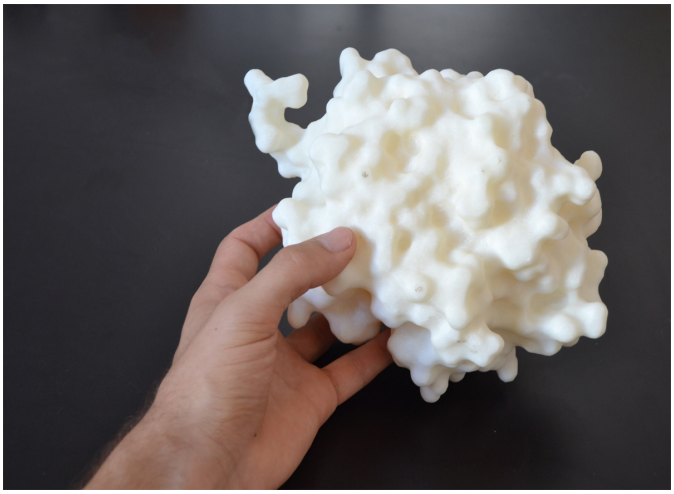


Figure 3.2: The molecular model held in hand

3.2 System calibration

As the goal of our application is to produce projections which blend in with a physical environment, we need to determine the physical properties of our system components: the tracking camera and the projector together with its attached camera have to be calibrated.

There were three main requirements to a calibration procedure. First of all, it had to be fast, because it has to be executed whenever there are slight changes in the setup. Then it had to be precise, because the accuracy of the whole system strongly depends on the calibration. Finally, it should be reasonable to be carried out by people who are not computer vision experts, as they are the target user group.

For these reasons, we opted for the method by Audet and Okutomi [AO09] to calibrate the projector together with its attached camera. After some research on existing approaches, we identified the method by Audet and Okutomi to be the most efficient and user-friendly one for calibrating projector-camera systems, and also the most suitable for our setup. With this method it is possible to align printed and projected markers in an interlaced manner, in contrary to the extension of Zhang's method [FHMF09], where the projected pattern must be located next to the printed one. This saves space, and as our working volume is very restricted, the use of a huge calibration plane would be problematic.

The independent infrared tracking camera can be either calibrated with the same application (but not simultaneously with the projector, as it is not able to perceive the projected patterns), or with any other single camera calibration procedure.

3.3 Tracking

As the 3D-printed molecular model should be freely moveable by a user, its position and orientation must be tracked in real-time.

Most of the SAR applications introduced in Section 2.3.4, which include object tracking, introduce some restrictions on this movement: the tangibles of the *ReacTable* and the *Tangible Tabletops* are equipped with markers at the bottom, which are tracked by a camera below the translucent table. Thus, the objects are bound to the plane of the table and the tracking is reduced to two dimensions. The optical tracking systems used for *Teegi* are capable of delivering six degrees of freedom tracking data, however, the system is designed such that the tangible elements are meant to remain in an upright position and on, or close to, the table surface. This helps to create an environment in which the visibility of all markers is assured at any time, especially if the markers are two-dimensional.

In our system, however, the augmented object should be freely rotatable by the user, which means that we have to account for unrestricted six degrees of freedom in manipulation.

We were concerned about the affordability of our system and therefore did not want to use expensive tracking devices, such as the *OptiTrack V120:Trio* ($3000) used for *Teegi* or a *Polhemus Fastrack* (a magnetic tracking system starting at $6000) used for the original *Dynamic Shader Lamps*. Therefore, we sticked to optical tracking using a consumer-level digital camera for the prototype. Initially, we aimed at implementing marker-less tracking, because markers have a negative impact on user interaction, as they might constrain manipulation or occlude parts of the tracked object. Furthermore, the usage of markers leads to increased setup costs, as markers have to be produced, attached to the object, and their transformation in relation to the object has to be determined.

Unfortunately, none of the marker-less tracking approaches we found during our research were appropriate for our purpose; they all operate on the detection of edge- or texture features. We, however, would need to track an SES model made of white plastic, which provides neither hard edges, nor textures. Therefore, we were restricted to a marker-based approach. We opted for a method related to the ArToolKit [KB99], which operates with two-dimensional printed patterns.

As mentioned above, our molecular model must be freely rotatable. Therefore, ideally, the markers must be attached to the object in a way, which ensures, that at least one marker is visible under every possible rotation. On the other hand, placing multiple markers all over the model would lead to unbearable occlusions. We made a compromise by mounting a marker cube on a stick attached to the model (see Figure 3.3). In this way, at least one side of the cube is visible under most rotations (as long as the cube is not occluded by the object), and no markers have to be attached to the model directly. This solution was also used by Gillet et al. [SWS+03].

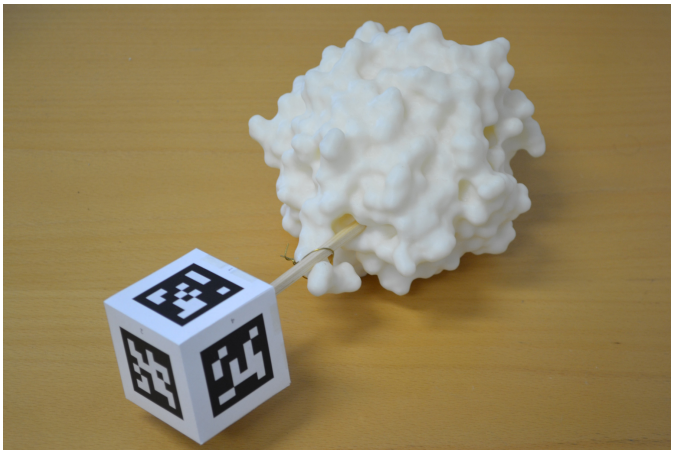


Figure 3.3: Molecular model with attached marker cube

We encountered a problem inherent to our setup: the optical marker detection can be disturbed by the light from the projector. If the marker is lit with inhomogeneous light sources, additional edges are introduced in the camera image and the marker detection algorithm of the ARToolkit fails.

However, we observed that a DLP projector hardly emits any infra-red light; so our solution consists in an infra-red tracking camera, additional to the one attached to the projector. In images delivered by such a camera, the printed markers are still visible (as long as they are illuminated by light sources that emit in the infra-red spectrum), while the light from the projector is not. As CCD/CMOS sensors are naturally very sensitive to infra-red light, a standard digital camera can easily be modified to operate in the infra-red spectrum by replacing the infra-red blocking filter present in such a camera by an infra-red-pass filter.

The emission spectrum of incandescent light bulbs contains relatively high amounts of infra-red light; enough to illuminate our scene sufficiently for infra-red marker tracking. However, this is not the case for many other types of light sources, such as fluorescent lamps or LED lamps. Furthermore, the projections appear much more intense and appealing in dark environments.

To tackle this additional problem, an additional infra-red light source can be used. When choosing such a light source, one has to make sure that it can evenly light the areas within the tracking camera's field of view. A uniform lighting is crucial for ARToolKit-based marker detection, because during the marker detection algorithm the image is binarized using a global threshold.

3.4 Transformations

In order to be able to use the tracking information to correctly register the virtual molecular model with the physical one, the transformations between the system components must be known. Figure 3.4 gives an overview.

The global coordinate origin of the system is given by a defined ARToolKitPlus marker (or a set of markers, to increase stability). As a camera views the origin, its own position and rotation with respect to the global coordinate frame can be determined. This information can then be used to express the transformations of tracked objects in global coordinates, rather than in camera coordinates:

$$\mathbf{M} = \mathbf{O}_{tc}^{-1}\mathbf{M}_{tc}$$

where $\mathbf{M}$ is the transformation matrix of the tracked object in world coordinates, and $\mathbf{O}_{tc}$ and $\mathbf{M}_{tc}$ are the transformations of the origin marker and the tracked object with respect to the coordinate frame of the tracking camera.

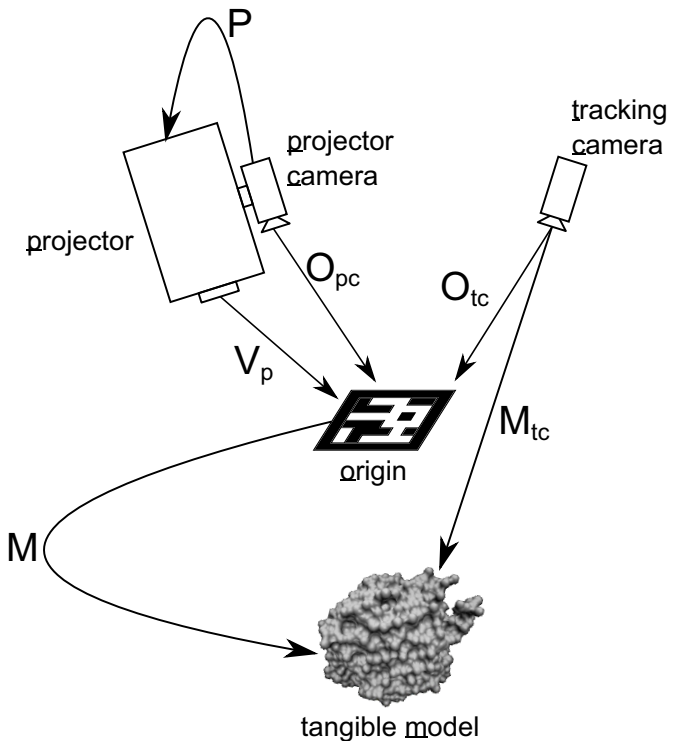


Figure 3.4: Rigid transformations between system components

The transformation $\mathbf{P}$ of the projector with respect to its attached camera is determined during calibration. As the attached camera also views the origin marker, the transformation of the origin with respect to the projector (or the projector's *view matrix*) $\mathbf{V}_p$ is given by:

$$\mathbf{V}_p = \mathbf{P}^{-1}\mathbf{O}_{pc}$$

where $\mathbf{O}_{pc}$ is the transformation of the origin marker with respect to the coordinate frame of the attached camera.

3.5 Rendering

Our application reads a surface mesh of the SES surface model of a complex molecule, including per-vertex color sets which encode various surface properties of the molecule. Those per-vertex colors are rendered directly. The color-sets can be switched at run-time to interactively change the visualization modality.

The images, which are projected onto the physical molecular model, are generated in a render window, that is displayed full-screen on the projector. To achieve the desired alignment of the physical and virtual object, the virtual molecule has to be rendered from the projector's point of view. The composition of the corresponding view matrix $\mathbf{V}_p$ was explained in Section 3.4. The required projection matrix is given by the internal parameters of the projector, which were acquired during calibration.

Our application does not aim at a realistic simulation of materials or lightning conditions, as other applications of *Shader Lamps* described in Section 2.3. Our approach mainly provides insight to color-coded data. Also, virtual lighting and shading, which is obligatory in screen-based rendering to produce a three-dimensional impression, is not necessary; the three-dimensionality is already provided by the 3D-printed model on which the visualization is displayed. In this case, shading effects would only distort the displayed information and confuse the user. Accordingly, the rendering for the projector output is reduced to a minimum and just passes through the interpolated per-vertex colors.

CHAPTER 4

Implementation

4.1 Hardware

In Chapter 3, the basic hardware setup was introduced. In the following, we will provide details about the individual components, which were used in our prototype implementation.

4.1.1 3D-printed molecular model

Our test data set consists of an SES representation of a complex protein: a haloalkane dehalogenase cocrystallized with NaI from a rhodococcus species [NPR+99], labeled as *1cqw* in the *Protein Data Bank* (PDB) [BWF+00]. It belongs to a family of enzymes that were extracted from bacteria living in areas contaminated by chemicals. Those enzymes are now extensively researched because they can help to degrade or detect chemicals in the environment and are also applicable to other areas like drug design. The size of the probe assumed to create the surface (see Section 2.1.1) corresponds to the size of a water molecule.

The model was 3D-printed in white acrylonitrile butadiene styrene (ABS) using a Stratasys Fortus 250mc 3D-printer. The maximum diameter is about 20 centimeters. The 3D-printing was carried out by our collaborators at the Masaryk University in Brno, Czech Republic.

In our prototype, only one model has been used. However, the presented system is designed in a way that it can also handle multiple molecular models to be used at the same time.

4.1.2 Digital projector

In principle, any projector can be used; however, as it has to be placed above the working surface, its weight and size should be considered. In our test setup we used a Samsung SP-P410M portable projector.

4.1.3 Digital camera

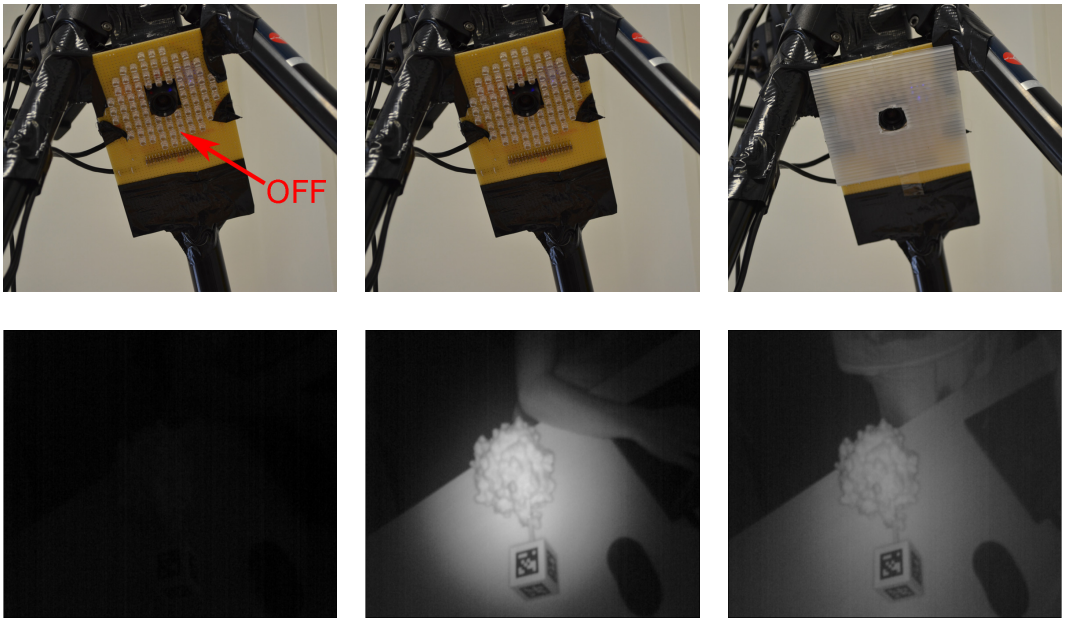
This camera is fixed to the projector and is mainly used to calibrate the projector. We used a standard digital camera that can be connected to the PC by USB. For computer vision applications a high frame rate is desirable; during our research we found that the PlayStation®Eye camera is a suitable solution. Depending on resolution and color mode, it can achieve frame rates of 60-120 Hz and is relatively cheap (meanwhile less than $10). It is originally an accessory for the Sony PlayStation®3, but there exists a Windows driver developed by Code Laboratories [cle10]. It comes with an API that enables developers to access device-specific parameters, use multiple PS-Eye cameras at the same time, and uniquely identify them.

4.1.4 Infra-red camera

The infra-red camera is used for optical tracking. The camera we used in our setup is also originally a PlayStation®Eye, but we modified it to respond to infra-red light only. This can easily be done by removing the built-in infra-red blocking filter and replacing it with an infra-red-pass filter (such as a few layers of black processed film). Since the camera is used for tracking, the frame rate and the possibility to set a short exposure time is crucial.

4.1.5 Infra-red light source (optional).

If the lighting environment does not provide sufficient amounts of infra-red light, or if the visual appeal of the projection should be improved by dimming visible light, the use of an additional infra-red light source is recommended. For our prototype, we built a custom USB-powered infra-red lamp, made of 80 IR-LEDs. The lamp is of circular shape and fits around the camera lens; thus we assure that the light always points in the view direction of the camera. As the used LEDs have a quite small angle of radiation of about 30°, a piece of multi-wall sheet was added in front of the LEDs to diffuse the light and achieve a more uniform light distribution within the field of view of the camera. Figure 4.1 shows the workspace only lit with fluorescent lamps, with direct IR-illumination and with an attached diffuser. With this setup, the system even works in dark environments (with low levels of visible background illumination), which makes the projected visualizations much more appealing.



(a) no IR-lighting (b) direct lighting (c) with diffuser

Figure 4.1: Comparison of different infra-red lighting setups

4.1.6 Personal computer

There are no special hardware requirements for the machine, as long as the GPU supports OpenGL version 3.1, or greater, and as long as there is an additional video output to connect a projector. In our test setup we used a Lenovo ThinkPad W520 notebook equipped with a NVIDIA Quadro 1000M graphics card.

4.2 Software

We implemented a standalone desktop application that incorporates tracking, the visualization of 3D molecular models and their correct projection on the tangible model, and provides a graphical user interface that enables the user to modify the projection output and monitor the involved devices. We used the programming language C++ and the Qt framework.

The only task we performed off-line with a third party application was the geometric calibration of cameras and projector.

4.2.1 System calibration

We calibrated the projector and its attached camera using the method of Audet and Okutomi [AO09]. We used the authors' Java-implementation of the algorithm and performed the calibration off-line. The results of the calibration are the internal parameters of the

camera and the projector, respectively, and the transformation between the camera and the projector.

For calibrating the independent infra-red tracking camera, we used a separate single-camera calibration procedure. We found the GML Camera Calibration Toolbox [gml13], which is based on OpenCV's implementation of Zhang's method, to be the most user-friendly freely available tool.

The obtained internal and external parameters then have to be transferred to the configuration files of our application.

4.2.2 Tracking

We opted for the tracking of printed markers using the ARToolKitPlus [WS07] library, because of its ease of use. The ARToolKitPlus, like many related libraries, uses printed two-dimensional patterns as markers. We extended the library by implementing a *Marker Set* entity, which consists of multiple markers and their spatial relations to each other. The poses of the individual markers delivered by the ARToolKitPlus are then interpolated to find the pose of the entity. This way, it is possible to build three-dimensional markers composed of two-dimensional patches, like the marker cube, that we attached to the 3D-printed model (see Section 3.3).

The tracking using the ARToolKitPlus was, out of the box, not very stable. Errors in the camera images led to noticeable jitter. To improve the tracking behavior, we implemented smoothing filters, such as an average filter and a Kalman filter for 6DOF tracking (see Section 2.5.2).

4.2.3 Rendering

The rendering was implemented in Qt's OpenGL environment. Our test data set consists of a surface mesh of the SES surface model of a complex molecule, including per-vertex color sets which encode various surface properties of the molecule. Those per-vertex colors are rendered directly. The color-sets can be switched at run-time using the graphical user interface (GUI) to interactively change the visualization modality.

We implemented one render window which is part of the GUI (see Section 4.2.4). It shows the virtual molecule rendered from one of the cameras' point of view, on top of the camera frame; it should thus be perfectly aligned with the tracked 3D-printed model. As mentioned above, the global origin is defined by a given marker, and a camera determines its own pose by viewing this marker. The transformation of the origin marker as seen from a specific camera ($\mathbf{O}_{tc}$ or $\mathbf{O}_{pc}$ in Figure 3.4) is equivalent to the view matrix required to render from that camera's point of view. The projection matrix is made from the internal camera parameters, which are determined during an offline calibration. The virtual model is rendered using a basic Phong shading.

The second (and more important) render window is displayed full-screen on the projector and produces the output that is projected onto the physical molecular model. To achieve the desired alignment of the physical and virtual object, the molecule has to be rendered

from the projector's point of view. To achieve this, the view matrix $\mathbf{V}_p$ (see Section 3.4) and a projection matrix given by the internal parameters of the projector are used. No lighting effects were used for shading, as this render window only aims at altering the surface coloring of a physical model.

4.2.4 The graphical user interface

The application can be controlled and monitored via a Qt-based graphical user interface (GUI), which is shown in Figure 4.2. It fulfills the following tasks:

Manipulation of the virtual models. For each visualized molecule, the visualization modality can be changed at run-time. In our example data set, the user can choose between visualizing atoms, charge, donors/acceptors, hydrophobicity or amino acid residues. The color-coding of those properties is shown in a legend beneath the modality drop-down menu. Furthermore, the transformation of the model with respect to the attached marker set can be modified, which allows for manual adjustments on the registration. The edited model configuration can be saved in a configuration file.

Render control. The rendering of the virtual objects can be disabled and enabled, and certain shader effects can be adapted at run-time.

Camera management. The user can switch through all connected cameras, watch their video streams (with the superimposed registered model and a visual representation of the global origin), and edit their parameters. While manipulating the physical model, the view from the tracking camera can provide a good hint about the working volume in which the model can be tracked.

Tracker management. The state of the trackers, including their frame rate and the number of currently detected markers, can be monitored. Furthermore, the user can chose between filtering modes for the tracking data.

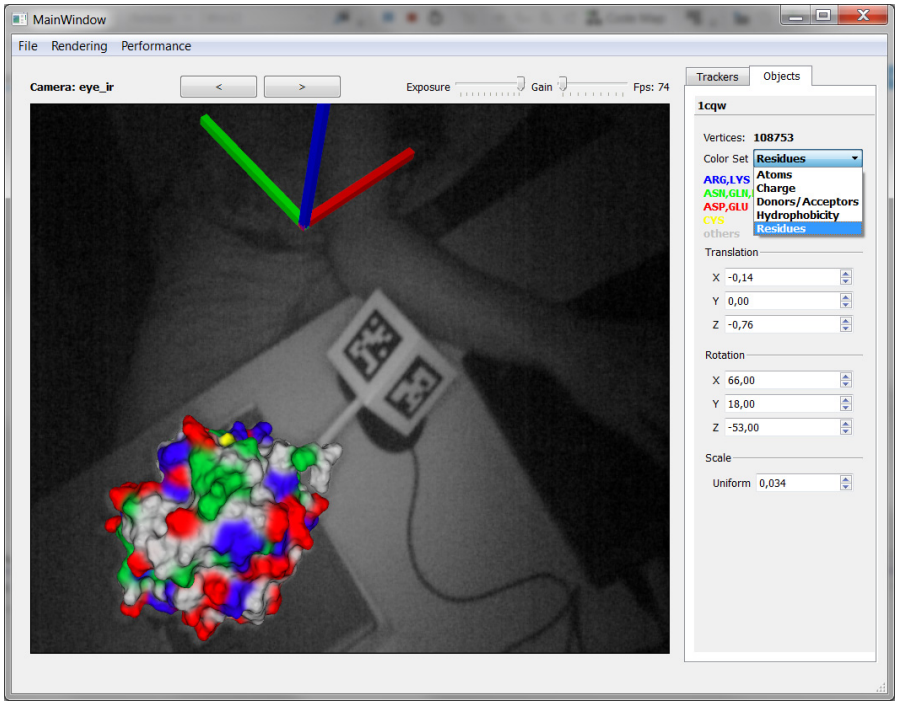


Figure 4.2: The graphical user interface

Now that we have described the hardware and software components of our prototype setup, the next chapter will review the system's functionality and performance.

CHAPTER 5

Results

Our implementation realizes the tangible molecular visualization system we were aiming for, with the exception of the marker-less pose estimation, which was replaced by an optical tracking of a mounted marker. The user can freely interact with the tangible molecular model, while color-coded molecular properties are correctly projected onto the surface. Depending on the current pose of the molecular model, the projected content is updated in real-time, such that it stays at the right place while the model is manipulated. Furthermore, the visualization modality can be switched at run-time, enabling the display of various molecular properties on a single tangible model.

5.1 Interaction

Figure 5.1 demonstrates our system in action. As mentioned above, the user can freely interact with the molecular model, while the projected visualizations stay correctly registered with the tangible surface. Usually, this interaction will include holding the object in one's hands, moving and turning it around, viewing it from different angles and feeling its structure. However, the model can also be statically placed on the table. For Figure 5.1a, one could imagine a group of students standing around the table where the molecule is presented. In Figure 5.1d, the object is placed on the table, while the user reviews additional material.

(a) (b) (c) (d) (e) (f) (g) (h)

Figure 5.1: Our system in action. The molecular model can be placed on a table or held in hand and viewed from different angles, while the projection stays registered with the tangible surface.

Furthermore, the different visualization modalities, which can be changed by the user at run-time, are shown. In Figures 5.1c 5.1d and 5.1g, a hydrophobicity map is visualized. The shown colors are interpolations between red (hydrophobic) and blue (hydrophilic). Another modality shows various amino acid residues distinguishable by different color encoding (Figures 5.1a, 5.1e, 5.1f). In Figure 5.1b donors are shown in brown, acceptors in green and regions that are both donors and acceptors in red. Figure 5.1h shows a visualization of the molecule's charge, where positively charged atoms are red, negative ones are blue and neutral regions are white. When a visualization modality is selected, the legend to the corresponding color coding is shown to the user in the GUI. In Figure 5.1h, the user takes a look at the GUI to find out what the projected colors mean.

Due to the hardware setup, the space, in which the molecular model can be freely moved, while the system keeps functioning, is limited. We call this space the *working volume*. In order to define the working volume of the system, two constraints have to be considered. First, the marker cube must be inside the view frustrum and focus range of the tracking camera. Second, the molecular model must be in the projection frustrum and focus range of the projector. Figure 5.2 depicts the working volume of our setup and an example for a valid position of the molecular model.

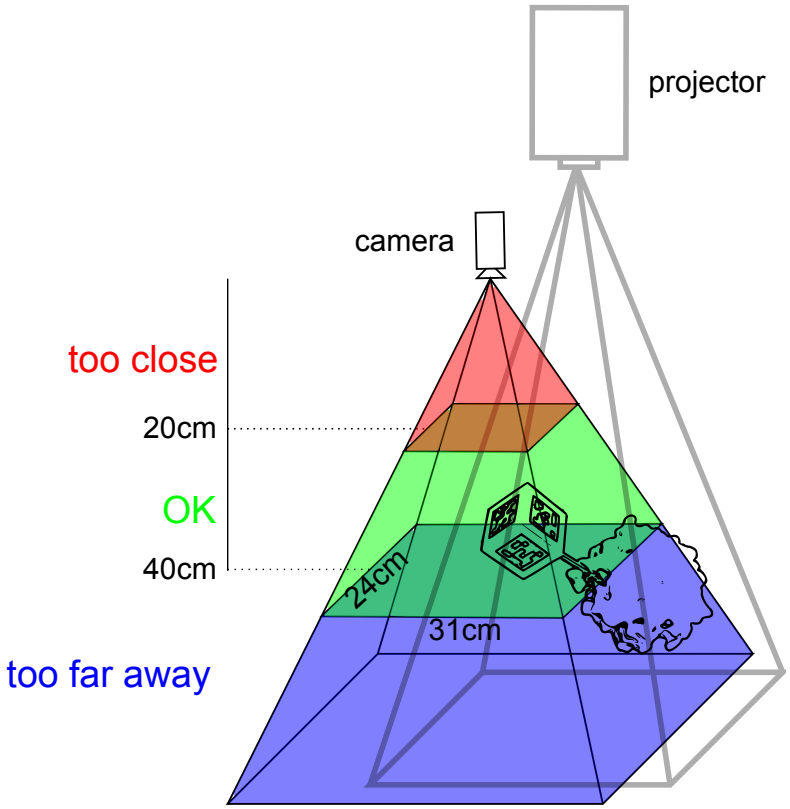


Figure 5.2: The constraints of the working volume and an example of a valid position of the model.

In our setup, the tracking camera is responsible for most of the restrictions. The infra-red modifications on the camera also had side effects on its optical properties. Objects are depicted sharply only up to a distance of about 30cm from the camera. Beyond that, the image gets increasingly blurry. We found that our markers (with a side length of 4cm) can be tracked up to a distance of 40cm from the camera.

Having a field of view of about 43° horizontally and 33° vertically, the marker can be moved within a rectangle of 31 × 24cm at a distance of 40cm from the camera. Also, the marker must not move too close to the tracking camera. At distances less than 20cm, the image gets overexposed due to the infra-red lamp attached to the camera.

The view frustrum of the projector, which is overlapping with the one of the camera, hardly restricts the working volume. Its field of view is slightly narrower than the camera's, but this can be compensated by placing the projector further away. The projector's focus range starts at a distance of about 55cm from the device. For our application, however, a slightly blurry projection is not problematic anyway, as the projected content does not contain hard edges.

5.2 Visualization

In our prototype, surface properties of the molecule are visualized via color coding. The set of shown properties can be changed at run-time, but are per se static. The corresponding legend can be found in the GUI.

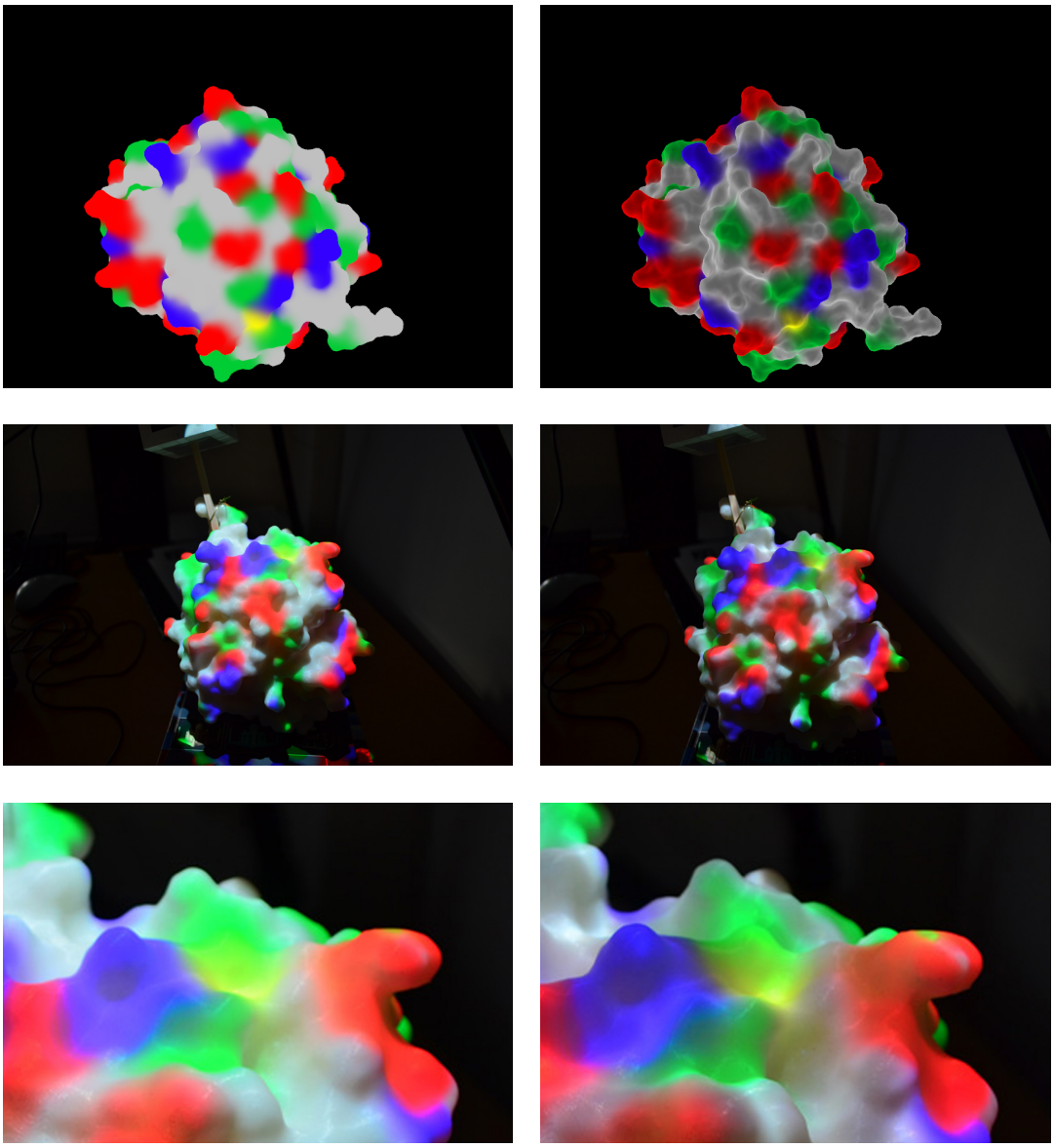
As we only use a single projector, the projection cannot cover the whole model. In Figure 5.1c, for example, it is obvious that only the upper half of the molecular model is reached by the projection. It is therefore necessary for the user to view the model from an angle similar to the direction of projection. In a setup like ours, where the user works on a table and the projector is mounted above, this does not pose a severe restriction to interaction. In everyday situations, we would also expect objects to be mainly lit from above, therefore inspecting objects from above is the most intuitive way anyway.

The aesthetic appeal of the visualization strongly depends on the ambient lighting situation. The darker the environment, the more intense and distinguishable the colors appear. A bright environment, on the other hand, leads to washed-out colors and a loss of contrast. Due to the infra-red lamp, that we equipped the tracking camera with, our visual tracking system is largely independent of the ambient lighting situation and can thus operate in environments with very low levels of visible light. The photographs of Figure 5.1 were taken in a half-light environment, at daytime with closed curtains.

Another point worth mentioning is the effect of the very material of the 3D-printout on the visualization. The material used for printing our model is slightly translucent and exhibits subsurface scattering effects. This leads to a glowing appearance of the augmented object and slightly blurred edges. If the effect is not desired, the model could be treated with matte paint. Parker et al. [PLS+15], for example, painted their models with diffuse gray paint spray, to obtain the most neutral results.

A slope-dependent intensity correction, as proposed by Raskar et al. [RWLB01] (see Section 2.3.2), was implemented and tested, but was felt to rather introduce confusion

than help to an aesthetically more pleasant result. As the SES model exhibits very small structures, incomprehensible effects occur with a small misalignment of the projection. Figure 5.3 shows a comparison of projection results with and without intensity correction.



(a) Without correction (b) With correction

Figure 5.3: Effects of slope dependent intensity correction. Top: render output. Middle: augmented object. Bottom: close-up.

5.3 Limitations

Our prototype exhibits certain limitations which could have a negative impact on its usability. The major limitations will be discussed in the following:

Working volume. Due to the hardware components of the prototype, the working volume is rather restricted. The user must take care to keep the marker cube inside the tracking camera's active volume (shown in green in Figure 5.2), in order to keep up the tracking and thus a correct projection. This additional task could draw the user's attention away from the visualized content.

Lag. In our prototype application, the average total time span between a real-world event and a corresponding update on the display device is currently about 120ms. We measured this by running a stopwatch on a monitor, capturing it with the digital camera, and displaying the camera image on the screen using our application. When taking a snapshot of the screen, the difference between the two visible clocks gives the lag. We found that about 2/3 of this lag is caused by the camera itself and 1/3 by our application overhead. For screen-based augmented reality, such as the camera view window of our GUI, the lag is hardly noticeable, because the rendered camera frame and the superimposed virtual content are equally late and stay registered. When augmenting with a projector, however, moving the tracked object with 0.1m/s (which corresponds to a moderate hand movement) already causes a projection offset of about 1.2cm, which is clearly noticeable.

Jitter. The optical tracking using the ARToolKitPlus was, out of the box, not very stable. Errors in the camera images led to noticeable and annoying jitter. To improve this, we experimented with smoothing filters, such as an average filter and a Kalman filter for 6DOF tracking (see Section 2.5.2). We achieved the most satisfying results by using a weighted average of the estimated poses of the last k frames. The results are still not perfect, as a compromise between smoothness and lag has to be made. The more frames are used, the smoother the tracking, but the more influence 'the past' has on the current tracker state, which leads to additional lag.

CHAPTER 6

Evaluation

After the implementation and internal testing of the prototype, we evaluated the applicability of our approach to tasks in the field of structural molecular biology, for which it was mainly designed. These were the main questions we wanted to answer in the course of the evaluation:

1. Is the proposed framework a useful tool for **research**?
2. Is the proposed framework a useful tool for **teaching/presentation**?
3. For which kinds of visualization tasks could it be used?
4. How does it compare to existing methods?
5. Are there usability issues?

We decided to perform an informal evaluation with a small group of domain experts from the fields of biochemistry and molecular biology, who are specialized on the structures of proteins and other biomolecules. We followed an exploratory approach, consisting mainly of the collection of verbal feedback in the course of live demonstration sessions of the system.

6.1 Setting

We conducted personal sessions with domain experts from the fields of biochemistry and molecular biology. Those sessions were filmed, in order to obtain a reliable documentation of the statements and comments of the experts. To reliably answer the main questions stated above, we set up a list of more fine-grained questions that we want to be answered during a session in one way or another:

- Usability:
  - What was the general impression? How did it feel?
  - Is the precision sufficient?
  - How problematic is the lag?
  - Did you notice other problems (e.g. color quality, hindrance by markers, working volume)?
- Suitability for research:
  - Does the application meet the requirements of researchers (e.g. their expectations stemming from existing frameworks)?
  - What is missing?
  - Do you see advantages of our system in comparison to screen-based solutions?
  - Can our setup improve understanding for experts (e.g. of the spatial structure)?
  - Which tasks do you think the system could be especially useful for? How should they be implemented?
- Suitability for teaching:
  - Are you involved in teaching? If yes:
  - Have you used 3D-prints for teaching?
  - Do you see potential in our approach for teaching?
  - Which molecular properties could be communicated more efficiently through our approach?

A study session with a domain expert was structured as follows: after a personal introduction, we gave an overview about our project, followed by a live demonstration of the system. Then, the expert could test the application him/herself, to get an idea of how it feels and reacts. The proband is encouraged to give comments and ask questions at all times. Those question from the list above, which were not answered while the expert was trying the system, were treated in a concluding discussion in the form of a semi-structured interview. We refrained from handing out our questions in the form of a printed questionnaire, because we recorded the session including all utterances anyway, and did not want to unnecessarily over-strain the experts' time.

Performing a larger scale quantitative user study was not feasible because of the following reasons: first, our target users, the domain experts described above, are a very small group of specialized people. It is thus extremely hard to find enough test subjects for a statistically significant quantitative evaluation. Second, as we were mainly interested in the usability and applicability of the system, such a study would have to include means to measure these traits. This is usually done by observing how fast and easily the probands can solve pre-defined tasks. However, there is a vast amount

of molecule related visualization tasks, depending on the exact specialization of every researcher. So at this point, we are still in the stage of exploring which tasks even come into consideration for being solved with our approach.

6.2 Realization

We conducted two separate evaluation sessions. The first one was held in our labs at the TU Wien, with a post-doctoral researcher from the Max F. Perutz Laboratories, a research and teaching facility for molecular biology (Figure 6.1a). She specialized on the analysis of the evolution of secondary structures in RNA and is currently involved in exploring the use of 3D-printing for molecular biology. It might be worth mentioning that the researcher also has a degree in the fine arts. The session lasted for about one hour and fifteen minutes and included, apart from the topics directly related to our project, a lot of discussion on the applicability of 3D-printed molecular models and molecular visualization in general.

The second session took place at the Loschmidt Laboratories, a group of the Masaryk University in Brno, Czech Republic, which is especially dedicated to protein engineering (Figure 6.1b). The participants consisted of a group of researchers from that facility, including the head of the department, the team leaders of the working groups of 'Environmental Chemistry and Ecotoxicology' and 'Biophysics', as well as two PhD students and two master's students. The very molecule which served as our test data set, was isolated and researched intensively in these laboratories. The session took about 40 minutes.

(a) TU Wien

(b) Loschmidt Laboratories, Brno

Figure 6.1: Evaluation sessions

6.2.1 Session at TU Wien

The expert tested the system using the visualizations of amino acid residues and hydrophobicity, which both exhibit bright and easily distinguishable color patches. Her first statement about the interaction was an expression of doubt, that a specific point at the surface of the model maintains the same color when rotating the object. She was confused because the colors seemed to be 'moving', partly due to the lag, partly due to jitter. She also found it problematic that the projection could not reach into holes

and cavities, because these are often the most interesting regions. She proposed to print models, which can be decomposed 'like puzzle pieces', which enables the exploration of such cavities. When trying to determine the meaning of the projected colors, she had to look on the GUI on screen for about five seconds.

In connection with the system's usability, we also talked about the format of input data. It was found, that molecular researchers are used to file formats containing semantic information about molecules, such as the positions and properties of individual atoms and structural information. An established standard for such files is the *.pdb* format, which can be found in the *Protein Data Bank* [BWF+00]. Our system, on the contrary, takes surface meshes as an input. This makes sense for our setup, because we need information of the exact surface of the 3D-printed model in our system to produce correct projections. But molecular researchers are not necessarily familiar with computer graphics methods and file formats. So, for our system to be accepted by the community, it would have to include a user-friendly way to transform a structural protein description into a surface mesh which can be 3D-printed and used as input for our visualization system.

The domain expert is currently evaluating the possibilities of 3D-printing for molecular biology, so we had a discussion about the general limitations of this medium and how to overcome them. According to her, one of the major problems of typical 3D-printed molecules is their lack of flexibility. Actually, when showing a molecule in a static way, a statistic average constellation is depicted. This may convey a wrong image, because proteins are flexible and dynamic, which is important for their functions. At that time, the expert was experimenting with printing molecules using flexible filaments, in order to make them deformable.

The second big problem with 3D printouts is, that they only show the surface of the molecule. The expert emphasized, though, that proteins are 'ensembles of structures'; by only viewing their surface and not their interiors, the spectator can find out only little about the structure of the molecule. In this context, again, it would make sense to print molecules decomposed into several 'puzzle pieces', to be able to access their interior.

The third problem the expert mentioned, was the static and often monochrome coloring of the printed models. For serious applications, she said, being able to see all of the molecule's properties (like charge, hydrophobicity, donors and exceptors, residues, or flexibility) is highly important. Actually, she has thought about solving this problem with projections before.

On the other hand she claimed that, in her experience, especially people who have worked on a particular molecular structure for several years, are very excited to be able to hold their subject of research in their hands in form of a 3D-printout; so she suspects that the affective aspects of 3D-printed proteins may exceed their actual usefulness.

Talking about possible application scenarios of our system, we agreed that for inter-molecular docking, it could have real advantages over purely virtual screen-based visualizations. Aligning two complex objects to each other in an interdigitating manner, such that they touch but not intersect, is extremely hard when working with a 3D visualization

program. It is rather trivial with tangible objects, because here the collision detection comes for free. The projected augmentations could aid this task in two ways: first, the visualized molecular properties can help to find a suitable docking site. Second, during a docking attempt, the touching points between the two molecules could be detected and visualized, still remaining visible for some time after taking the objects apart. This way one could see where exactly the molecules were touching. However, the domain expert again emphasized that the flexibility of proteins plays an important role, also for docking. During a docking process, not only the features, but also the structure itself may change, which brings us back to one of the inherent limitations of rigid 3D printouts.

The expert also saw potential in projecting animated content, instead of static surface properties. The dynamic nature of the protein could be suggested by introducing random movements to the projected properties, depending on the flexibility of the corresponding surface patch. With animations, the tangible model could also support a presentation, when 'telling a story' about the molecule. There could be transitions between different interconnected molecular properties, or regions of interest could be highlighted.

According to the expert, being able to project internal structures would be a great improvement to our system, because one of the major limitations of 3D-printed molecular models is that they can only show the surface.

Regarding the application context of our system, the expert mentioned the two domains of visualization: analyzing data versus presenting data. She said that our approach is probably not useful for analyzing data. In general, two-dimensional visualizations are more suitable for this field. On the other hand, our system might be a great tool for presentation, because it is simply exciting and attracts the attention of the audience.

This also corresponds to her evaluation of the system in terms of applicability for research and education. Molecular researchers are mostly specialized on one particular molecule and 'speak about different things', than those that can be shown with an augmented 3D-printed model. Thy need much more detailed information, mostly on specific sub-parts of the molecule. At most, our approach could be useful for interdisciplinary communication, or presenting research results to a broad audience.

This leads to its applicability for education, which was rated positive by the expert, despite her concerns that, because all of the limitations mentioned above, 3D-printed models could invoke wrong ideas in students. On the other hands, she sees great potential for installations at science museums or science fairs, like the *Ars Electronica* [ars15].

6.2.2 Session at Loschmidt Laboratories, Brno

To make the following subsection more readable, we will introduce the following abbreviations for the protagonists of this session:

- JD...head of the department
- JB...head of Biophysics
- ZP...head of Environmental Chemistry and Ecotoxicology

JD was the first expert to test the system. He immediately commented, that the tracking is not working properly, because the marker cube attached to the model was outside the tracking camera's field of view. His next comment was already on the fact, that he is excited to hold this protein in his hands for the first time. This confirms the presumption of the domain expert we interviewed in Vienna, that scientists enjoy physical models of their research subjects. He concluded by saying that he likes the system because he is able to hold the molecule in his hands and touch it, but the visualization must become more robust, in order to be useable; the performance of the tracking should not be that location-dependent.

When JB was trying the system, he confirmed, that it is difficult to keep the marker cube inside the tracking volume, because one has to have a second eye on the camera stream shown in the GUI of our application. He also noted, that the marker cube obscures the projection, if it is between the projector and the molecular model. Also the self-shadowing of the model was problematic for him, and he concluded, that it would be good to have more than one projector and blend their images, in order to reduce the surface areas which are not reached by the projection. He also commented, that the projection 'sometimes jumps completely', when the marker is out of focus and the jitter thus increases.

Furthermore, JB pointed out the issue of orientation on the object surface. When using classic desktop visualizations programs, he is used to the display of annotations, like residue numbers, which help to determine which part of the molecule one is currently looking at. He says that it is therefore difficult to connect our tangible visualization to his understanding of the protein. It would therefore make sense to include annotations in our projected visualization. JD agreed on that.

In general, when using desktop visualization programs, the experts are hardly viewing the molecular surface only, because the display of inner structures is an aid for orientation as well. So if the surface is displayed, it is mostly transparent. The experts attending our session, for example, were mostly concerned with tunnels in the proteins. Through those tunnels, smaller molecules can enter the protein and react with it [BJG+15]. Our system, however, does not provide any means to observe those tunnels or even show their entrances on the surface.

This directly leads to the main disadvantage of our system in the eyes of the attendees: only the surface can be displayed. ZP noted, that to him, the most important information about a molecule can be found in its interior. It would therefore be favorable to print models which can be opened. But in general, he would prefer purely virtual, hologram-like 3D visualizations over an actual tangible model, because then one would have unlimited possibilities of looking inside the molecule, by displaying cross-sections or making it transparent. In this context, we were discussing the possibility of using the tangible model as a mere controller and performing the visualizations on screen or in a holographic way, similar to the work of Gillet et al. [SWS+03] [GSSO05]. JB commented that, for such an approach, it would still be nice to have information projected on the model for orientation.

On the other hand, JB pointed out, that the surface is also an essential part of the molecules. There are research groups, for example, that focus on protein-protein interactions, for which the surface is essential. JD agrees, that protein-protein interactions are a promising application scenario for our approach.

The experts agreed, that the real advantage of our system, in comparison to just printing multiple versions of the same protein in different colorings, is the potential ability to display animations. JD proposed to simulate the flexibility of the protein by rendering movements on the surface of the molecule, depending on the flexibility of the corresponding part.

One of the attending PhD students commented, that visualizing DNA binding sites, for example on a ribosome, would be a nice application. The dynamic movements of the DNA on the surface could be animated.

JB was especially interested in the visualization of ligand passages, because this is a special focus of his work. Ligands are small molecules, which can enter a protein through its tunnels, undergo a conversion and eventually exit the molecule. This process could be animated with our system by projecting the pathways of the ligands to the tunnels on the protein surface. JD commented, that also the ligand molecules themselves could be visualized, as they move on the protein's surface and disappear in the tunnels. In this context, again, being able to take the 3D-printed model apart would make sense. This way, the ligand could first be visualized on the surface and, when entering a tunnel, the protein model could be taken apart and the processes happening in the interior could be animated.

When the experts were asked, if they see potential for the use of our approach in research, JB said that it might be useful for initially familiarizing oneself with a new structure, as the global shape could be perceived more easily with a tangible model. Another advantage of the tangible model might be the intuitive perception of structural properties, like the relative distance between certain parts of the surface.

JD, on the other hand, commented that researchers are so used to their desktop visualizations, that he finds it difficult to judge how our approach would perform when doing 'real work'.

On the application of teaching, JD noted that the system is for sure interesting and attractive, but would have to be more robust and easy to use for classical teaching situations. The professor wants to come to class five minutes before the lesson starts and be able to start teaching, instead of spending half an hour on the setup and calibration of the system. Otherwise, it would be perhaps more feasible to print a few multi-colored versions of the same molecule and pass them to the students. On the other hand, doing this on a regular basis would be quite expensive and would not support animations. ZP, who is also involved in teaching, said that he would probably also prefer the 'simple way' of printing several multi-colored models. It should be noted, that none of the attending researchers have actually used 3D-printed models in teaching before, but they would be very interested in doing so.

JB said, that he sees the greatest potential of our system in public demonstrations for lay persons. It could be used at universities' open days or education fairs, to encourage people to study molecular biology; there it might have large impact because it looks so impressive. Other suitable occasions could be science fairs or beginner's days for students.

However, he concluded that for being used by experts, the system is probably too complicated and 'superficial', in the truest sense of the word.

6.3 Results

Considering the feedback collected from the domain experts during our evaluation sessions, we can answer our initial main questions as follows:

Is the proposed framework a useful tool for research?

Although it could be used for certain tasks, like familiarizing oneself with a new protein or intuitively exploring properties like relative distances, its possibilities are generally too limited to be really useful for researchers. They are either concerned with very specific parts or properties of a protein, which cannot be displayed with our method, or they require other representations than the surface alone; either because they are per se interested in inner structures and tunnels, or they need those structures for orientation on the surface.

Is the proposed framework a useful tool for teaching/presentation?

The participants of the evaluation sessions agreed that our approach shows great potential to be used for teaching applications. The setup is probably too complicated for use in classic frontal teaching situations, but could be utilized in seminars and discussion groups to interactively rely content. This could be especially effective if the system was extended with animations, which can support 'telling stories' about a protein.

The setup could be especially useful for addressing lay audiences. Its exciting and appealing nature makes it perfectly fit for installations at science fairs, open house presentations and similar events.

For which kinds of visualization tasks could it be used?

The biggest potential was anticipated for animated visualizations, because these could never by accomplished by 3D-printing alone, even when using multi-color prints. Possible visualization scenarios include showing the trajectories of ligands, conveying the flexibility of the molecule by rendering vibrations, tell interactive stories by changing between modalities or highlighting parts of the molecule, and animating DNA binding sites.

Another group of applications that emerged were protein-protein interactions, such as intermolecular docking. Here the user can benefit from the tangible properties of our setup.

How does it compare to existing methods?

One advantage of our system in comparison to screen-based approaches is, naturally, the tangible component. It makes it easier to obtain a quick overview of the structures and aids applications like inter-molecular docking. From the psychological point of view, the system's attractive and exciting appearance makes it a more effective medium for broad audiences.

The system's disadvantages in comparison to screen-based approaches are all connected with the static nature of the 3D-printed model. In our case, only the surface can be shown, whereas inner structures might be even more meaningful to specialists. Furthermore, proteins are actually flexible, which is also difficult to convey using a rigid model.

Are there usability issues?

As we anticipated, the major issues were the restricted working volume, lag and jitter. These are problems that can be attributed to the tracking system. Furthermore, the testers were missing annotations or other unique markers; without them, they have problems orienting on the surface and relating their understanding of the protein to our representation. Another fact which was criticized was, that not all parts of the molecular model's surface could be reached by the projections.

Another usability-related issue that emerged was the need for a simple workflow to convert semantic molecular data to surface meshes, which can be loaded into our system and also be used for 3D-printing.

CHAPTER 7

Conclusion and Future Work

7.1 Synopsis

In this thesis, I presented an approach for tangible molecular visualization, driven by the idea that tangible interfaces improve data insight and understanding. Tangible models of complex molecules can now be easily produced with 3D-printing technology. Those static monochrome models can then be augmented with digital light projections, to visualize the molecular properties of interest directly on the surface of the physical model. Using knowledge about the geometry of the molecular model, the optical properties of the digital projector and the exact spatial relation between projector and model, the visualization content can be rendered in a way, such that it is perfectly registered with the physical model.

We developed a prototype system which implements the projection of color-coded molecular properties onto a tangible 3D-printed model. Equipped with a real-time optical tracking system, the position and rotation of the tangible model is updated continuously, enabling the user of the system to interactively explore the molecule. Within a defined working volume, the model can be freely moved an rotated while the projected content stays registered with the physical object. The user can interactively change the set of visualized molecular properties at run-time.

The prototype system was evaluated by a group of domain experts from the fields of biochemistry and molecular biology. We found that our system had an impressive and appealing effect on spectators. It is thus very suitable for communicating research results to a broad audience, like in exhibitions or museums, especially when the projected visualizations are animated. For being used by scientists on a daily basis, whether for teaching or research, the system in its current state is too complex to use and, due to the nature of the rigid surface models used, not flexible enough.

7.2 Conclusion

The work on this thesis was an insightful trip into the realms of spatial augmented reality, molecular visualization and 3D-printing. As we developed the described visualization system from scratch, many difficulties had to be overcome.

The mathematical background required to render an image which, when displayed on a projector, perfectly fits a real-world object as a texture, turned out to be surprisingly trivial. It came down to applying a (somewhat mind-twisting) concatenation of transformation matrices to a 3D mesh representation of the real-world object. In our case, such a mesh was readily available, because the 3D-printout was created from it.

The hard part was to actually obtain those transformation matrices, which incorporate the geometric properties of projectors and cameras as well as the spatial relations between the relevant system components. These are properties of the real world, and thus have to be measured, which leads to the inherent problem of measurement errors, that impact the performance of the system.

Some of these measurements can be performed off-line, because the values are not expected to change over time, like the intrinsic parameters of projectors and cameras, and the spacial relations between fixed system components, such as the transformation between a projector and a camera attached to it. These parameters can be determined with elaborate calibration procedures, which can be tedious, but in general deliver precise results. However, that those values are expected to be constant, does not mean that they actually are. Even small mechanical influences can lead to changes in extrinsic or intrinsic parameters, and require a re-calibration of the system.

The other part of the measurements has to be performed in real-time. Throughout this work, we called that process the *tracking* of the tangible model. Fast and precise professional tracking systems—no matter on which physical principle they operate—were far beyond the budget of this thesis. So, for our prototype, we had to implement a tracking system ourselves. We refrained from implementing the initially anticipated marker-less object tracking, because, for our target class of objects, it turned out that the implementation would deliver enough material for another dedicated master's thesis. But even the marker-based approach that we used as a substitute turned out to be insufficient for our application. First of all, it was too slow. A delay of a tenth of a second is hardly noticeable in screen-based augmented reality applications, because the camera frame and the virtual augmentation are displayed equally delayed; in our case, however, an offset between tangible object and projection was visible. Second, it was not stable enough. The images delivered by our consumer level cameras were too noisy for extracting jitter-free tracking data. In hindsight, our self-implemented tracking system was determined to be inferior to a several-thousand-Euro professional solution. It would have been wise to acknowledge that fact from the beginning and either try to get (possibly temporal) access to a professional system or lower one's expectations.

These technical difficulties and the resulting loss of usability surely contributed to the opinion of some reviewing domain experts, that our approach might be too unstable and error-prone to be used on a daily basis. Furthermore the evaluation interviews revealed

the fact, that only in rare cases a surface representation alone can satisfy the needs of a researcher. And although the system could be extended with a view-dependent rendering of inner structures, those interiors would not be tangible; therefore their display would not bring any advantages in comparison to a screen-based visualization. The restriction to surfaces however, is not a property of our system, but of the medium of 3D-printing as such. From these observations, we conclude that molecular researchers print molecular models mainly for pleasure, rather than for scientific reasons.

This aspect of pleasure, however, opens a field of applications for our system, that we did not anticipate when starting the project. The visual and haptic appeal of our system was uniformly evaluated as being aesthetically pleasant, impressive and interesting. It is therefore very well suited to communicate to attract the attention of a broad audience and stimulate their interest for molecular research results. If we had this realization earlier in the development phase, we could have tailored our prototype to fit this context of application even better.

If we were to name one major insight, that can be taken away from the work on this project and the related research, it would be this: augmented reality is, despite the attention it lately receives, still in its childhood, and a lot of work has to be done until it is ready for broad application.

7.3 Future work

Although the main goals of our project were achieved, there is a lot of space for improvements and extensions to our system.

One of the biggest challenges lies in the improvement of the tracking. As the molecular model does not exhibit textures or edges, which are used as detectable features in state-of-the-art marker-less rigid object tracking approaches, other ways must be tread. The most promising approaches we encountered during our research was a tracking based on the silhouette of the model, as the silhouette of an SES molecular model is very distinct from each perspective. One could for example render the object's silhouette from all possible view angles (in discrete intervals) and try to register them with the object's silhouette extracted from a camera image. Assuming that the object is only rotated by a small amount between two succeeding frames, only 'neighboring' rotations could be searched to speed up the process. Alternatively, the use of other tracking modalities could be investigated. From all of the tracking technologies introduced in Section 2.5, we believe magnetic tracking to be the most desirable for our application. It is very precise and fast, and the sensors can be quite small an thus be placed on the model inconspicuously. Furthermore no direct line of sight between sensor and base station is required. The working volume of such a device is restricted by a maximum distance between sensors and base station, which is typically about one meter. This is more than sufficient for our purpose. The downsides of the approach are the high costs of the devices, and the magnetic field distortions caused by metallic objects, other electronic devices and their wiring.

The printed model itself is, according to biochemists, not optimal as well. The form of a protein is not constant over time; we are just depicting the state with the maximum likelihood. Also during docking processes, proteins change their form. One suggestion was to print the models out of flexible material; this is possible with today's 3D printers. On the other hand, when deforming a flexible object, this would add another challenge to the tracking, as the projection would have to follow this deformation. Also the inner structures of the molecule, such as tunnels, cannot be seen in a model such as ours. One solution would be to decompose the molecule into parts, like puzzle pieces, which are tracked individually. They could then be taken apart and put back together by users to reveal the inner structures.

According to feedback from domain experts, support for animations would be a reasonable extension to our system. Not only time-dependent changes, but also inter-molecular effects could be visualized. For example, when attempting to find a configuration for docking between two molecules, the surface patches, where the two molecules were touching, could be highlighted. This could provide feedback about the quality of the docking configuration.

Another improvement could be made by implementing view dependent effects. If the position of the user's eyes are known, structures which are not congruent with the model's surface, such as inner structures, could be shown via projection. In order to correctly render such scenes, the two-pass algorithm presented in Section 2.3.1 would have to be applied. This approach would also be interesting for other fields of application. One could, for example, visualize medical computed tomography (CT) data by 3D-printing a body part and augmenting it with a projection of its interior, using volume rendering methods.

Concluding, it should be emphasized that the presented approach is not restricted to the visualization of proteins, for which it was initially designed. The method of printing three-dimensional objects and using them as a display surface could also be explored for other areas of scientific visualization and computer graphics. For example, we envision the application of our approach for surface flow visualizations on models of cars or airplanes or view-dependent volumetric visualizations of computed tomography data on models of bodies and body parts.

Bibliography

[AO09] S. Audet and M. Okutomi. A user-friendly method to geometrically calibrate projector-camera systems. *Computer Vision and Pattern Recognition Workshop*, 0:47–54, 2009.

[ars15] Ars Electronica.
http://www.aec.at/news/, September 2015.

[B+00] J. Brickmann et al. Molecular Graphics – Trends and Perspectives. *J. Mol. Model.*, 6(6):328–340, February 2000.

[BB07] Javier Barandiarán and Diego Borro. Edge-based markerless 3d tracking of rigid objects. In *ICAT*, pages 282–283. IEEE Computer Society, 2007.

[BJG+15] Jan Byška, Adam Jurčík, Meister Eduard Gröller, Ivan Viola, and Barbora Kozlíková. Molecollar and tunnel heat map visualizations for conveying spatio-temporo-chemical properties across and along protein voids. *Computer Graphics Forum*, 3(34):1–10, May 2015. EuroVis 2015 - Conference Proceedings.

[Bou08] J. Y. Bouguet. Camera calibration toolbox for Matlab, 2008.

[BR05] Oliver Bimber and Ramesh Raskar. *Spatial Augmented Reality: Merging Real and Virtual Worlds*. A. K. Peters, Ltd., Natick, MA, USA, 2005.

[Bra00] G. Bradski. OpenCV. *Dr. Dobb's Journal of Software Tools*, 2000.

[BWF+00] H M Berman, J Westbrook, Z Feng, G Gilliland, T N Bhat, H Weissig, I N Shindyalov, and P E Bourne. The Protein Data Bank. *Nucleic acids research*, 28(1):235–242, 2000.

[Cat13] R Catanese. 3D Architectural Videomapping. XL(September):2–6, 2013.

[cle10] CL-Eye Platform.
https://codelaboratories.com/products/, 2010.

[CM10] Sara Comai and Davide Mazza. Haptic and visual rendering for multimodal exploration of molecular information. In *Proceedings of the 24th BCS*

Interaction Specialist Group Conference, BCS '10, pages 221–229, Swinton, UK, UK, 2010. British Computer Society.

[CP53] Robert Corey and Linus Pauling. Molecular models of amino acids, peptides, and proteins. *The Review of Scientific Instruments*, 24:621–627, 1953.

[DH12] Peter Dalsgaard and Kim Halskov. Tangible 3d tabletops: Combining tangible tabletop interaction and 3d projection. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design*, NordiCHI '12, pages 109–118, New York, NY, USA, 2012. ACM.

[Eve01] Cass Everitt. Projective texture mapping, 2001.

[FDS12] Peter Ferschin, Monika Di, and Angelo Stefan. Projected Time Travel : Architectural Heritage Projection in Situ. pages 1–9, 2012.

[FGF+14] Jérémy Frey, Renaud Gervais, Stéphanie Fleck, Fabien Lotte, and Martin Hachet. Teegi: Tangible EEG Interface. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 301–308, New York, NY, USA, 2014. ACM.

[FHMF09] G. Falcao, N. Hurtos, J. Massich, and D. Fofi. Projector-Camera Calibration Toolbox, 2009.

[FMSS14] Matthias Faessler, Elias Mueggler, Karl Schwabe, and Davide Scaramuzza. A Monocular Pose Estimation System based on Infrared LEDs. *Icra*, 2014.

[GB78] Jonathan Greer and Bruce L. Bush. Macromolecular shape and surface maps by solvent exclusion. *Proceedings of the National Academy of Sciences of the United States of America.*, 75(1):303–307, 1978.

[gml13] GML Camera Calibration Toolbox. *http://graphics.cs.msu.ru/en/research/projects/3dreconstruction/cppcalibration*, 2013.

[GSSO05] Alexandre Gillet, Michel Sanner, Daniel Stoffler, and Arthur Olson. Tangible Interfaces for Structural Molecular Biology. *Structure*, 13(3):483–491, March 2005.

[HCB12] Michael S. Horn, R. Jordan Crouser, and Marina U. Bers. Tangible interaction and learning: The case for a hybrid approach. *Personal and Ubiquitous Computing*, 16(4):379–389, 2012.

[HDS96] William Humphrey, Andrew Dalke, and Klaus Schulten. VMD – Visual Molecular Dynamics. *Journal of Molecular Graphics*, 14:33–38, 1996.

[his13] The History Blog. *http://www.thehistoryblog.com/archives/25193*, May 2013.

[HvDG94] Kenneth P. Herndon, Andries van Dam, and Michael Gleicher. The challenges of 3d interaction: A chi '94 workshop. *SIGCHI Bull.*, 26(4):36–43, October 1994.

[HZ03] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003.

[JGAK07] Sergi Jordà, Günter Geiger, Marcos Alonso, and Martin Kaltenbrunner. The reactable: Exploring the synergy between live music performance and tabletop tangible interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, TEI '07, pages 139–146, New York, NY, USA, 2007. ACM.

[JSM+14] Brett Jones, Rajinder Sodhi, Michael Murdock, Ravish Mehra, Hrvoje Benko, Andrew Wilson, Eyal Ofek, Blair MacIntyre, Nikunj Raghuvanshi, and Lior Shapira. Roomalive: Magical experiences enabled by scalable, adaptive projector-camera units. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 637–644, New York, NY, USA, 2014. ACM.

[Kaj86] James T. Kajiya. The rendering equation. *SIGGRAPH Comput. Graph.*, 20(4):143–150, August 1986.

[Kal60] Re E Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME-Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[KB99] H. Kato and Mark Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proceedings of the 2nd International Workshop on Augmented Reality (IWAR 99)*, San Francisco, USA, October 1999.

[KBE09] Michael Krone, Katrin Bidmon, and Thomas Ertl. Interactive visualization of molecular surface dynamics. *IEEE Trans. Vis. Comput. Graph.*, 15(6):1391–1398, 2009.

[KKL+15] Barbora Kozlíková, Michael Krone, Norbert Lindow, Martin Falk, Marc Baaden, Daniel Baum, Ivan Viola, Julius Parulek, and Hans-Christian Hege. Visualization of biomolecular structures: State of the art. *EuroVisSTAR2015*, pages 061–081, May 2015.

[kur11] Vimeo: Bacardi Together Mapping Projection at the Kursalon, Vienna. *https://vimeo.com/23359503*, May 2011.

[LC87] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21(4):163–169, August 1987.

[LK93] Susan J. Lederman and Roberta L. Klatzky. Extracting object properties through haptic exploration. *Acta Psychologica*, 84(1):29–40, 1993.

[LK03] Susan J. Lederman and Roberta L. Klatzky. Touch. *Handbook of Psychology, Part two: Sensory Processes*, page 147–176, 2003.

[lon04] London Science Museum. *http://www.sciencemuseum.org.uk/images/I053/10321094.aspx*, January 2004.

[LSFK10] João Paulo Lima, Francisco Simões, Lucas Figueiredo, and Judith Kelner. Model Based Markerless 3D Tracking applied to Augmented Reality. *SBC Journal on 3D Interactive Systems*, 1:2–15, 2010.

[LWF11] P. Lincoln, G. Welch, and H. Fuchs. Continual surface-based multi-projector blending for moving objects. In *Virtual Reality Conference (VR), 2011 IEEE*, pages 115–118, March 2011.

[Mar07] Paul Marshall. Do tangible interfaces enhance learning? In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, TEI '07, pages 163–170, New York, NY, USA, 2007. ACM.

[MGV+14] Annette Mossel, Georg Gerstweiler, Emanuel Vonach, Klaus Chmelina, and Hannes Kaufmann. Vision-based long-range 3D tracking, applied to underground surveying tasks. *Journal of Applied Geodesy*, 8(1):43–63, 2014.

[MK12] Ashok Kumar M and Sahityapriyadharshini K. A survey of artoolkitbased augmented reality applications, 2012.

[MWP06] Simon Meers, Koren Ward, and Ian Piper. Simple , Robust and Accurate Head-Pose Tracking Using a Single Camera. *Baseline*, (NaturalPoint), 2006.

[NPR+99] Janet Newman, Thomas S. Peat, Ruth Richard, Lynn Kan, Paul E. Swanson, Joseph a. Affholter, Ian H. Holmes, John F. Schindler, Clifford J. Unkefer, and Thomas C. Terwilliger. Haloalkane dehalogenases: Structure of a Rhodococcus enzyme. *Biochemistry*, 38(49):16105–16114, 1999.

[O'R87] Joseph O'Rourke. *Art Gallery Theorems and Algorithms*. Oxford University Press, Inc., New York, NY, USA, 1987.

[PK07] Thomas Pintaric and Hannes Kaufmann. Affordable infrared-optical pose tracking for virtual and augmented reality. In Gabriel Zachmann, editor, *IEEE VR Workshop on Trends and Issues in Tracking for Virtual Environments*, pages 44–51, Aachen, 2007. Shaker Verlag. Vortrag: IEEE Virtual Reality 2007, Charlotte, NC (USA); 2007-03-14 – 2007-03-17.

[PLS+15] Min Ki Park, Kyu Je Lim, Myoung Kook Seo, Soon Jong Jung, and Kwan H. Lee. Spatial augmented reality for product appearance design evaluation. *Journal of Computational Design and Engineering*, 2(1):38–46, 2015.

[PMK13] Antoine Petit, Éric Marchand, and Keyvan Kanani. Augmenting markerless complex 3d objects by combining geometrical and color edge information. In *IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2013, Adelaide, Australia, October 1-4, 2013*, pages 287–288, 2013.

[PPL+99] Steven Parker, Michael Parker, Yarden Livnat, Peter pike Sloan, Charles Hansen, and Peter Shirley. Interactive ray tracing for volume visualization. *IEEE TRANSACTIONS ON COMPUTER GRAPHICS AND VISUALIZATION*, 5, 1999.

[PPP06] Frank L. Pedrotti, Leno M. Pedrotti, and Leno S. Pedrotti. *Introduction to Optics (3rd Edition)*. Benjamin Cummings, 3 edition, April 2006.

[RBF01] Ramesh Raskar, Deepak Bandyopadhyay, and Henry Fuchs. Dynamic shader lamps: Painting on movable objects. In *Proceedings of the IEEE and ACM International Symposium on Augmented Reality (ISAR'01)*, ISAR '01, pages 207–, Washington, DC, USA, 2001. IEEE Computer Society.

[Ric77] F. M. Richards. Areas, Volumes, Packing, and Protein Structure. *Annual Review of Biophysics and Bioengineering*, 6(1):151–176, 1977.

[Roe11] Kevin Roebuck. *3D Printing: High-impact Emerging Technology - What You Need to Know Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*. Tebbo, 2011.

[RQPR+06] Monica Rubio, Arturo Quintana, Hebert Pérez-Rosés, Ricardo Quirós, and Emilio Camahort. Jittering reduction in marker-based augmented reality systems. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3980 LNCS:510–517, 2006.

[RWLB01] Ramesh Raskar, Greg Welch, Kok-Lim Low, and Deepak Bandyopadhyay. Shader lamps: Animating real objects with image-based illumination. In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques*, pages 89–102, London, UK, UK, 2001. Springer-Verlag.

[SB08] Erk Subasi and Cagatay Basdogan. A new haptic interaction and visualization approach for rigid molecular docking in virtual environments. *Presence: Teleoper. Virtual Environ.*, 17(1):73–90, February 2008.

[SFC+11] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *Proceedings of the 2011 IEEE Conference on*

Computer Vision and Pattern Recognition, CVPR '11, pages 1297–1304, Washington, DC, USA, 2011. IEEE Computer Society.

[SIS04] Steven Schkolne, Hiroshi Ishii, and Peter Schroder. Immersive design of dna molecules with a tangible interface. In *Proceedings of the Conference on Visualization '04*, VIS '04, pages 227–234, Washington, DC, USA, 2004. IEEE Computer Society.

[SWS+03] Ganesh Sankaranarayanan, Suzanne Weghorst, Michel Sanner, Alexandre Gillet, and Arthur Olson. Role of haptics in teaching structural molecular biology. In *Proceedings of the 11th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTICS'03)*, HAPTICS '03, pages 363–, Washington, DC, USA, 2003. IEEE Computer Society.

[UES01] Matthew Uyttendaele, Ashley Eden, and Richard Szeliski. Eliminating ghosting and exposure artifacts in image mosaics. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'2001)*, volume II, pages 509–516, Kauai, Hawaii, December 2001. IEEE Computer Society.

[WF02] Greg Welch and Eric Foxlin. Motion tracking: No silver bullet, but a respectable arsenal. *IEEE Comput. Graph. Appl.*, 22(6):24–38, November 2002.

[Wil78] Lance Williams. Casting curved shadows on curved surfaces. *SIGGRAPH Comput. Graph.*, 12(3):270–274, August 1978.

[WS07] Daniel Wagner and Dieter Schmalstieg. ARToolKitPlus for Pose Tracking on Mobile Devices, 2007.

[Zha00] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1330–1334, November 2000.

[Zha04] Z. Zhang. *Camera Calibration*, chapter 2, pages 4–43. G. Medioni and S.B. Kang, eds., Prentice Hall Professional Technical Reference, 2004.