

Graphbasierte UI-Konzepte zur intuitiven Anpassung einer 3D-Rekonstruktionspipeline

Bachelorarbeit im Studiengang Medieninformatik

Patrick Creutzburg



Hochschule für Technik und Wirtschaft
Fakultät Informatik / Mathematik
Friedrich-List-Platz 1
01069 Dresden

1. Betreuer

Prof. Dr. rer. nat. Marco Block-Berlitz

2. Betreuer

Dr. phil. Hendrik Rohland

April 2021

Inhaltsverzeichnis

1	Einleitung	1
2	Theorie und verwandte Arbeiten	3
2.1	3D-Rekonstruktion	3
2.1.1	Anwendungsfall Archäologie	3
2.1.2	<i>Archaeo3D</i> -Videogrammetriesoftware	5
2.2	Usability	5
2.3	User Experience	8
2.4	Psychologischer Blick auf Design	9
2.4.1	Kommunikation Mensch und Computer	10
2.4.2	Farbwirkung und Farbpsychologie	11
2.4.3	Intuition und intuitives Interface	12
2.5	Mathematischer Hintergrund	14
2.5.1	Graphentheorie	14
2.5.2	Bézierkurven	15
2.6	Technologie der Umsetzung	16
2.6.1	React	17
2.6.2	Electron	17
3	VisualArchaeo3D	18
3.1	Beschreibung des Nutzungskontexts	18
3.2	Anforderungen an die UI	18
3.3	Design der Anwendung	19
3.3.1	Grundaufteilung	19
3.3.2	Farben, Icons und Typographie	20
3.3.3	Mockup	21
3.4	Umsetzung der UI	22
3.4.1	Allgemeiner Prozessknoten	23

3.4.2	Settings-Container	25
3.4.3	Die Hauptkomponente	26
3.4.4	Quellknoten	27
3.4.5	Frame-Extraction	29
3.4.6	Ausführungs- und Kontrolleinheit	30
3.4.7	VisualArchaeo3D-Testdurchlauf	32
4	Usability-Test	34
4.1	Zielsetzung	34
4.2	Methode	34
4.2.1	Probandenvorstellung	34
4.2.2	Begründung des Studiendesigns	35
4.2.3	Durchführung	36
4.3	Darstellung und Auswertung der erhobenen Daten	37
4.3.1	Auswertung der Use-Cases	37
4.3.2	Auswertung der Schlussbefragung	40
4.4	Diskussion der Ergebnisse	43
5	Zukünftige Bedeutung graphbasierter und intuitiver Interfaces	44
	Literatur	46
	Anhang	48
	Eigenständigkeitserklärung	73

Abbildungsverzeichnis

1	Nutzeroberfläche der graphbasierten Software <i>Blender</i>	2
2	Die selektive Farbauslöschung des Wassers	4
3	Die Radiale Verzeichnung am Beispiel eines Rechtecks	5
4	Unterschiede und Gemeinsamkeiten von UX und Usability	8
5	Das RASMUSSEN-Modell der menschlichen Kognition (nach HEINECKE)	11
6	Darstellung des Assoziationsraums zwischen Betrachter und Design .	13
7	Veranschaulichung eines ungerichteten und eines gerichteten Graphens	15
8	Eine Bézierkurve mit Stütz- und Kontrollpunkten	16
9	Übersicht über die grundlegende Aufteilung von <i>VisualArcheo3D</i> . .	20
10	Farben und Icons von <i>VisualArcheo3D</i>	21
11	Das Mockup von <i>VisualArcheo3D</i>	22
12	Die finale Realisierung von <i>VisualArcheo3D</i>	23
13	Der Settings-Container	25
14	Visualisierung des Quellknotens	28
15	Der Frame-Extraction-Knoten	29
16	Die Executer-Komponente	31
17	Demonstration der Funktionalität von <i>VisualArcheo3D</i>	33
18	Demonstration der Funktionalität von <i>VisualArcheo3D</i>	33
19	Die Antwortverteilung pro Frage der Use-Cases	38
20	<i>VisAWI-S</i> Auswertungsdiagramm	41
21	<i>SUS</i> -Auswertungsskala	42
22	<i>SUS</i> -Auswertungsdiagramm	42
23	Vorschläge zur Verbesserung der Usability	43

1 Einleitung

Software ist längst kein Bereich mehr, in dem sich ausschließlich Experten¹ bewegen. Sie findet Einzug in immer neue Sparten und ist auch für fachfremde Nutzer ein integraler Bestandteil des Alltags geworden. Prominente Beispiele dafür sind, neben vielen anderen, Industrie 4.0 oder auch die voranschreitende Digitalisierung in Schulen und Universitäten. Nicht zuletzt deshalb steigen die Anforderungen an die Nutzeroberfläche. Um Software einem breiteren Publikum anbieten zu können, muss diese von einer Vielzahl von Menschen mit unterschiedlichem technischen Hintergrund und Erfahrungsschatz bedienbar sein. Dazu müssen mögliche Einstiegsschwellen auf ein Minimum reduziert oder wenn möglich, vollständig eliminiert werden.

Ein weiterer bedeutender Faktor in der Nutzung von Software ist Zeit. Die eigene Erfahrung zeigt, dass man die Zeit, um die Bedienung einer Software zu erlernen, selten hat, oder schlicht nicht aufbringen möchte. Gerade im professionellen Sektor ist der Faktor Zeit von besonderer Bedeutung, da eine effizientere Nutzung einer Anwendung beispielsweise das Senken von Kosten zur Folge haben kann.

Während zur Pionierzeit von Software noch die reinen Funktionen einer Applikation im Vordergrund standen, sehen sich Softwareentwickler heute vor neuen Herausforderungen. Um konkurrenzfähig zu sein, müssen sich Entwickler die Frage stellen, wie leicht oder schwer sich Nutzer in ihrer Anwendung zurecht finden. Sie müssen die „Usability“ ihrer Software optimieren. Um das Ziel einer guten Usability zu erreichen, sollte der Mensch selbst, insbesondere die Eigenarten in seiner Wahrnehmung, genauer betrachtet werden. Um eine leicht zugängliche Anwendungsoberfläche zu erschaffen, die den Menschen in ihren Fokus stellt, muss sich ein Entwickler fragen:

„Lieber Nutzer, wer bist Du und wie tickst Du?“

Auf der zwischenmenschlichen Ebene findet ein Großteil der Kommunikation auf einer Art Metaebene statt. Der Mensch hat die Möglichkeit sein Gegenüber, intuitiv, anhand von Mimik, Gestik, Stimmfall und anderen subtilen Signalen zu „lesen“. Die Vermutung liegt nahe, dass ebendieses Verhalten auch bei der Mensch-Maschine-Interaktion zur Anwendung kommt. Darum gilt es herauszufinden, welche subtilen Signale einer Software Einfluss auf die individuelle, subjektive Wahrnehmung des Menschen haben. Versteht ein Softwareentwickler die menschliche Kommunikation und Wahrnehmung, verschafft ihm das in der Entwicklung seines Produktes einen großen Vorteil und er kann barrierefreie Anwendungen erstellen, die einem breiten Publikum zugänglich sind.

¹Im Folgenden wird aus Gründen der besseren Lesbarkeit ausschließlich die männliche Form benutzt. Es können dabei sowohl männliche als auch weibliche Personen gemeint sein.

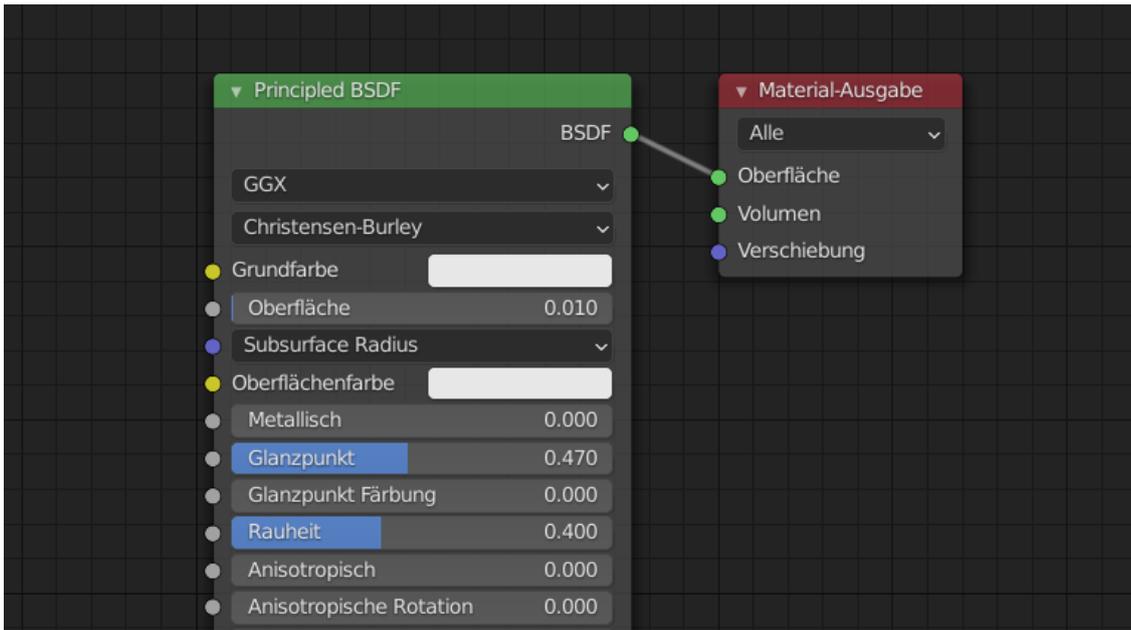


Abbildung 1: Die 3D-Animationssoftware *Blender* nutzt bereits Graphen zur Konfiguration von Oberflächen- und Renderparametern.

Im Zentrum der Arbeit steht das Konzept einer potentiellen Nutzeroberfläche für die Videogrammetriesoftware *Archaeo3D*. Um die Verarbeitungspipeline der Software individualisierbar zu machen, schlägt diese Arbeit ein graphbasiertes Konzept vor, wobei auch die menschliche Intuition eine zentrale Rolle spielt. Ähnliche Bedienkonzepte (siehe Abb. 1) werden bereits erfolgreich in 3D-Animations- oder Videoeditingsoftware genutzt. Bekannte Beispiele sind unter anderen *Blender*² und *DaVinci Resolve*³.

Mittels dieser praktischen Umsetzung soll die Frage beantwortet werden, ob eine graphbasierte Realisierung im Anwendungsfall der 3D-Rekonstruktion, speziell der Videogrammetrie, sowohl zweckdienlich als auch intuitiv ist.

Um dieser Frage nachzugehen, werden zuerst die notwendigen theoretischen Grundlagen vorgestellt. Nach der Vorstellung des UI-Konzepts wird dessen intuitive Bedienbarkeit mittels eines Usability-Tests evaluiert.

²<https://www.blender.org/>

³<https://www.blackmagicdesign.com/products/davinciresolve/>

Alle im Text angegebenen URLs sind gültig zum Zeitpunkt der Abgabe.

2 Theorie und verwandte Arbeiten

Der vorliegende Abschnitt bietet eine Übersicht über die wichtigsten theoretischen Grundlagen dieser Arbeit. Dazu wird zuerst die 3D-Rekonstruktion im Kontext der Archäologie vorgestellt. Da intuitive Bedienbarkeit eng mit der menschlichen Wahrnehmung verwandt ist, werden im Anschluss auch psychologische Aspekte betrachtet. Der Abschnitt endet mit einem Überblick über den mathematischen Hintergrund des Projektes.

2.1 3D-Rekonstruktion

Rekonstruktion im Allgemeinen ist das Nachbilden oder Wiederherstellen des ursprünglichen Zustandes eines Objektes. 3D-Rekonstruktion beschreibt den Vorgang, aus gemessenen Daten der echten Welt, ein dreidimensionales Computermodell abzuleiten. Dabei geht es, besonders im Bereich der Archäologie, darum, eine möglichst akkurate Nachbildung des abzubildenden Objektes zu erstellen.

Digitale Nachbildungen spielen im wissenschaftlichen Kontext eine immer größer werdende Rolle und dienen beispielsweise der Messung von Größen und Längen oder dem Nachbilden zum Zwecke der Konservierung. Im Gegenteil zur analogen Rekonstruktion, die beispielsweise im Modellbau angewandt wird, zeichnet sich die digitale Herangehensweise durch eine weitaus höhere Flexibilität aus. In einer Software können Modelldetails mit minimalem Aufwand verändert, hinzugefügt oder entfernt werden [17].

Die 3D-Rekonstruktion verwendet Photo- oder Videomaterial, um die Lage von Objekten im Raum zu kartographieren. Dazu nutzt sie Mechanismen der *Computer-Vision*, um vollautomatisch 3D-Modelle aus dem zuvor erstellten Material abzuleiten.

2.1.1 Anwendungsfall Archäologie

Das Rekonstruieren stellt für Archäologen in ihrer wissenschaftlichen Arbeit eine der relevantesten Aufgaben dar. 3D-Rekonstruktion bietet den Forschenden vielversprechende Möglichkeiten das Forschungsobjekt zu verarbeiten und auszuwerten. So macht sie es unter anderem möglich, sogar noch im Anschluss an eine Expedition am heimischen Schreibtisch Messungen und Nachforschungen zu betreiben. Oftmals muss aus wenigen historischen Überresten ein Modell abgeleitet werden, um zu veranschaulichen, wie ein Objekt im ursprünglichen Zustand aussah [17].

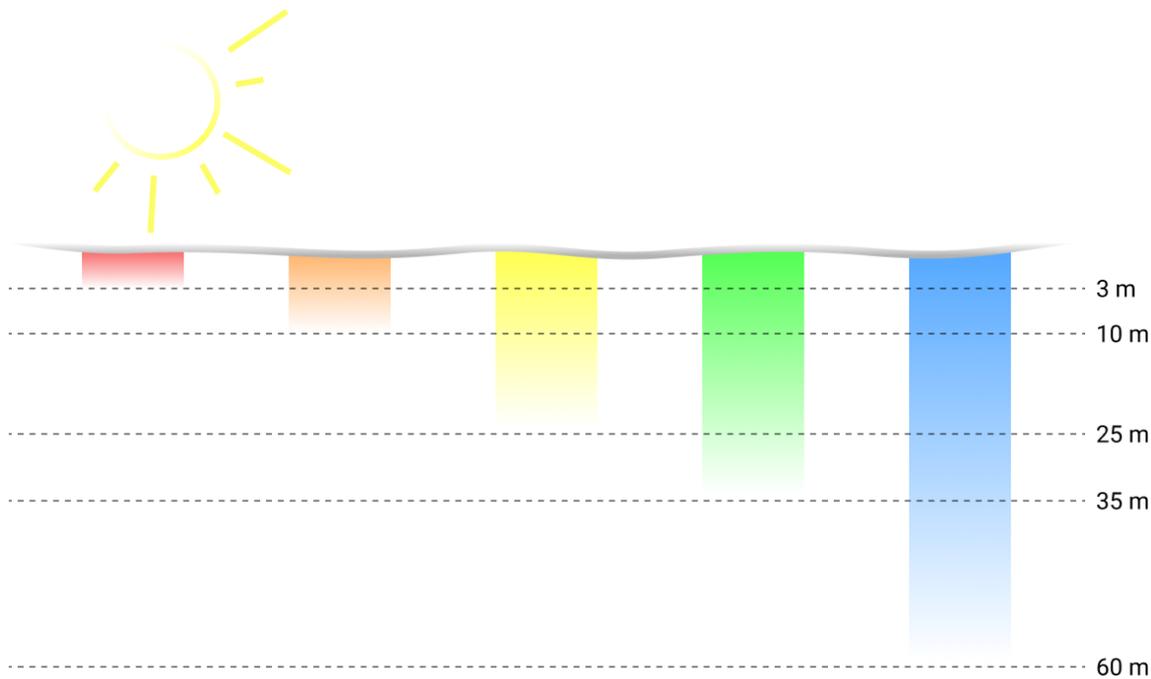


Abbildung 2: Die selektive Farbauslöschung des Wassers sorgt dafür, dass bei steigender Tiefe sukzessive Farben des Lichtspektrums verloren gehen (Abbildung in Anlehnung an [4]).

Insbesondere im Anwendungsfall der Archäologie kann der videogrammetrische Ansatz sein hohes Potential ausspielen. Er ist kosteneffektiv, schnell und einfach anzuwenden, zudem robust und fehlerunanfällig [5].

Heutzutage nutzen Archäologen bei Ausgrabungen oftmals *unmanned aerial vehicles (UAVs)*, um Bildaufnahmen von Objekten zu erzeugen, die anschließend mittels Software weiterverarbeitet werden. Dazu müssen aus dem fortlaufenden Bildmaterial die zur Erstellung eines 3D-Modells benötigten Einzelbilder, die Keyframes, extrahiert werden.

Archäologen erforschen bei ihren Expeditionen auch Objekte, die sich Unterwasser befinden. Bei Tauchgängen mit zunehmender Tiefe muss bei der Weiterverarbeitung des Videomaterials zudem die selektive Farbauslöschung des Wassers (siehe Abb. 2) berücksichtigt werden, um ein optimales Ergebnis zu erzielen [4].

Da reale optische Systeme nicht perfekt sind und einen Punkt im Objektraum nicht 1:1 auf einen Punkt im Bildraum abbilden, entstehen verschiedene Abbildungsfehler. Diese Aberrationen führen nicht zu einer Bildunschärfe, sondern rufen geometrische Verzerrungen in der Abbildung hervor.

Ein typischer Abbildungsfehler, besonders bei Luftaufnahmen, ist die *Radiale Verzerrung* (siehe Abb. 3). Diese wird durch eine Abweichung des realen Linsensystems von der Zentralprojektion verursacht und zeichnet sich durch die „[...] tonnen-

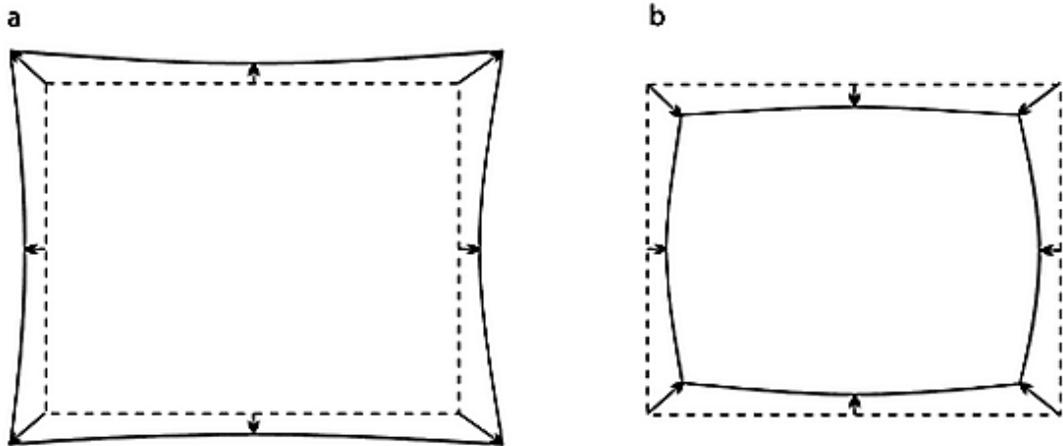


Abbildung 3: Ein Rechteck erscheint durch die Radiale Verzerrung gekrümmt. **a** positiv oder kissenförmig verzerrt; **b** negativ oder tonnenförmig verzerrt. (Abbildung aus [15])

oder kissenförmigen Verzerrungen von Quadraten [...]” [15] aus. Dadurch erscheinen gerade Linien gekrümmt (ebd.).

Für eine möglichst genaue Rekonstruktion archäologischer Objekte sollte daher der Fehler durch die Radiale Verzerrung berücksichtigt werden.

2.1.2 *Archaeo3D*-Videogrammetriesoftware

Archaeo3D implementiert eine semiautomatische 3D-Rekonstruktionspipeline zum Steuern und Optimieren des Rekonstruktionsprozesses. Diese beinhaltet einen Algorithmus zur automatischen Selektion von Keyframes zur Erstellung von 3D-Rekonstruktionen aus fortlaufendem Bildmaterial. Dabei kombiniert sie Technologien wie beispielsweise *VisualSFM*⁴ oder *CMPMVS*⁵ [16], um automatisiert die am besten für die 3D-Rekonstruktion geeigneten Bilder eines Videostreams zu extrahieren und anschließend gegebenenfalls farblich zu korrigieren und von Radialer Verzerrung zu bereinigen [5] .

2.2 Usability

Usability ist ein Teil der User Experience (UX). Sie ist ein stets aufgabenbezogenes Qualitätsmerkmal, das die Gebrauchs- und Nutzerfreundlichkeit eines Produktes beschreibt. Anwender erleben die Usability einer Software hauptsächlich über ihre Nutzeroberfläche (UI). Weil moderne Softwarenutzer vor allem im Berufsleben einer

⁴VisualSFM: <http://ccwu.me/vsfm/>

⁵CMPMVS: https://www.cs.cmu.edu/~reconstruction/basic_workflow.html

zunehmend größeren Menge an Daten ausgesetzt sind und diese in möglichst wenig Zeit bearbeiten müssen, ist es notwendig, dass Interfaces anwenderzentriert gestaltet werden. Eine intuitive Gestaltung hilft den Nutzern ihre Ziele effizient, effektiv und zufriedenstellend zu erreichen. Der Anwender kann durch eine gute Usability sofort erkennen, welchen Zweck UI-Elemente haben und wie er sie zielgerichtet einsetzen kann [23]. Anhand des folgenden Alltagsbeispiels wird das Wirken einer guten Usability deutlich:

Ein Kunde bucht mithilfe einer App einen für ihn zeitlich passenden Termin beim Friseur. Sein zuvor festgelegtes Ziel wird vollständig erreicht (*Effektivität*). Durch das Verwenden der App hat er nur die Hälfte an Zeit gebraucht, die er für einen Anruf benötigt hätte. Zeit und Aufwand für die Buchung werden durch die App im Vergleich zum Anruf verringert (*Effizienz*). Nach der Terminbestätigung ist der Kunde begeistert von der unkomplizierten Abwicklung der Terminvereinbarung. Die Bedürfnisse und Erwartungen des Nutzers wurden durch den positiven Eindruck der App sogar übertroffen (*Zufriedenstellung*) [9].

Die Usability bedient sich psychosozialer Aspekte menschlichen Denkens, um ihr Ziel der Nutzerfreundlichkeit zu erreichen und geht damit in der Gestaltung der UI weit über die Argumente bloßer Ästhetik hinaus. Weil der Mensch mit seinen Bedürfnissen im Vordergrund steht, sorgt die Usability dafür, dass die Anwendung lernt, den Menschen zu verstehen und nicht umgekehrt. Hat der Nutzer sein Ziel in einer angemessenen Form erreicht, ist die Usability der Software als anwenderfreundlich zu bewerten [14].

NIELSEN liefert mit seinen zehn Heuristiken einen entscheidenden Baustein für das Entstehen des modernen Usability-Designs. Diese Heuristiken sind als grobe Richtschnur zu verstehen, die dabei helfen soll, eine Software möglichst ergonomisch zu gestalten. Im Folgenden werden seine Gedanken aufgeführt und erläutert [20]:

1. **Sichtbarkeit des Systemstatus** | *Visibility of System Status*:

Durch (zeitlich) angemessenes Feedback soll der Nutzer darüber informiert werden, in welchem Zustand sich die Anwendung aktuell befindet.

2. **Übereinstimmung von System und Realität** | *Match between System and the Real World*:

Das Anwendungsdesign soll Logiken und Konzepte aus der realen Welt des Nutzers verwenden, sodass er diese leicht versteht.

3. **Kontrolle des Nutzers und Freiheit in der Nutzung** | *User Control and Freedom:*

Nutzer sollen zu jeder Zeit die Kontrolle über die Anwendung behalten, um beispielsweise eine versehentlich gewählte Aktion abbrechen zu können.

4. **Konsistenz und Standards** | *Consistency and Standards:*

Die Sprache und die Benennung eines Elementes innerhalb der Anwendung soll nicht variieren.

5. **Fehlervorbeugung** | *Error Prevention:*

Ein gutes Design soll Probleme des Anwenders antizipieren und ihnen vorbeugen.

6. **Erkennen statt Erinnern** | *Recognition Rather Than Recall:*

Alle verfügbaren Elemente, Aktionen und Optionen sollen dem Nutzer jederzeit ersichtlich sein.

7. **Flexibilität und Effizienz in der Nutzung** | *Flexibility and Efficiency of Use:*

Die Anwendung soll mehrere Wege zu einem Ziel anbieten, sodass Laien sowie erfahrene Nutzer gleichermaßen effektiv arbeiten können. Routinierten Anwendern soll beispielsweise eine Steigerung der Effizienz ermöglicht werden, indem sie Shortcuts verwenden.

8. **Ästhetisches und minimalistisches Design** | *Aesthetic and Minimalist Design:*

Irrelevante Informationen sollten nach Möglichkeit vermieden werden.

9. **Hilfe beim Erkennen, Diagnostizieren und Beheben von Fehlern** | *Recognize, Diagnose, and Recover from Errors:*

Fehlermeldungen sollen nachvollziehbar formuliert sein und bei der Fehlerbehebung helfen.

10. **Hilfe und Dokumentation** | *Help and Documentation:*

Das bestmögliche Design braucht keine Hilfestellung, dennoch ist es sinnvoll, Nutzern durch eine verständliche Dokumentation dabei zu unterstützen, ihre Aufgaben zu erfüllen.

Diese Heuristiken fußen auf wissenschaftlichen Erkenntnissen der Hirnforschung, sind jedoch nicht universell. Zusätzliche Usability-Tests sind daher unerlässlich, um eine hohe Qualität der Usability zu gewährleisten [14].

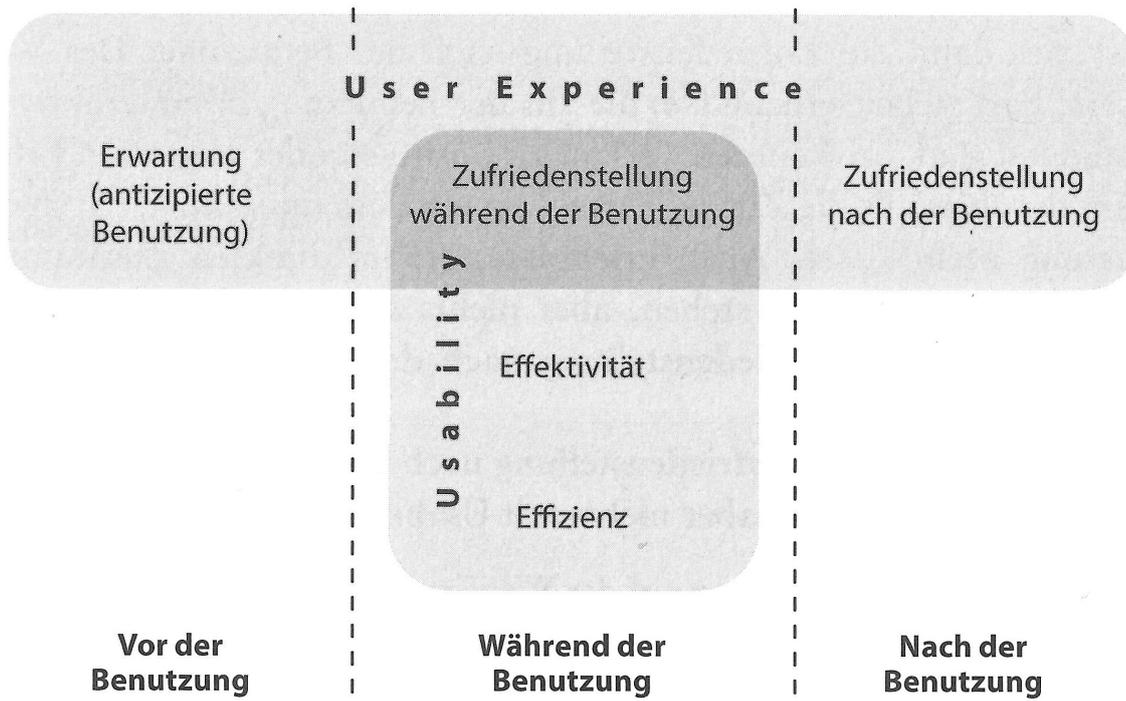


Abbildung 4: UX und Usability haben eine Schnittmenge in der Zufriedenstellung des Anwenders während der Nutzung. Die UX beeinflusst den User jedoch auch vor und nach der Nutzung und ist somit weitreichender als die Usability (Abbildung aus [9]).

2.3 User Experience

Die UX sieht die Anwendung im Gesamtkontext und ist damit weitreichender als Usability allein. Während sich Usability vorwiegend mit der Effizienz und Effektivität des Interfaces während der Nutzung beschäftigt, betrachtet die UX alle Berührungspunkte, die der Nutzer mit der Software oder ihrem Anbieter haben kann. Dabei fragt sie, welchen subjektiven Eindruck der Nutzer von der ihm vorliegenden Anwendung hat. Sie möchte den Anwender auf einer emotionalen Ebene zu erreichen, mit dem Ziel ihn zu binden und zu begeistern [14].

Während die UI das Nutzererleben nur während der Benutzung beeinflusst, hat die UX auch vor und nach dem Gebrauch der Software Einfluss auf das Erleben des Anwenders (siehe Abb. 4). Sie ist an Wertevorstellungen, Erwartungshaltungen und Vorlieben des Nutzers geknüpft und hat seine Zufriedenstellung zum Ziel [9].

Die User Experience eines interaktiven Systems wird durch verschiedene Faktoren beeinflusst. Zu ihnen zählen beispielsweise: „*Markenimage, Präsentation [...], Preis, Funktionalität, Usability, psychische und physische Fähigkeiten des Nutzers sowie seine Erfahrung, Einstellungen und Persönlichkeit, der Nutzungskontext sowie die Servicedienstleistungen rund um das interaktive System*“ (ebd.).

Viele dieser Einflüsse wirken auf den Nutzer und Konsumenten unbewusst. Dazu gehören unter anderem das Design, die Emotionen des Anwenders bei der Verwendung aber vor allem auch die Marke des Produkts. Psychologische Untersuchungen zeigen, dass gerade dieses Markenerleben nicht nur mit rationalen Ursachen wie dem praktischen Nutzen und einer einfachen Bedienbarkeit begründet werden kann, sondern emotionale Aspekte auch einen starken Einfluss darauf haben. Eine Marke ist deshalb nicht nur als Logo zu verstehen, sondern als Gesamtheit aller Eindrücke die ein Anwender von ihr hat [18].

Ein prominentes Beispiel für die Wirkung des Markenimages auf die User Experience ist das Unternehmen *Apple*. Um sich in einem hart umkämpften Markt mit vielerlei Konkurrenzprodukten, die vergleichbare Leistungsmerkmale aufweisen, einen Marktvorteil zu verschaffen, legt es besonderes Augenmerk auf das geschickte in Szene setzen seiner Produkte.

Diese Strategie kommt insbesondere dann zum tragen, wenn ein Kunde sich zwischen zwei Produkten mit ähnlichen Spezifikationen entscheiden muss. Er wählt dann eher das Produkt, das ihn emotional besser erreicht (ebd.). Der Kognitionswissenschaftler NORMAN beschreibt die Wirkung von Emotionen auf unser Denken folgendermaßen:

„Emotionen sind untrennbar und ein wichtiger Bestandteil von Kognition. Alles was wir tun, alles was wir denken ist emotional eingefärbt, vieles davon unbewusst. Emotionen verändern wiederum die Art wie wir denken und dienen als konstanter Leitfaden für angemessenes Verhalten, sie leiten uns weg vom Schlechten hin zum Guten“ [22].

2.4 Psychologischer Blick auf Design

Die Wirkung eines Designs auf den Menschen ist stets individuell, aber keineswegs willkürlich. Seine Wahrnehmung beruht auf den psychosozialen Mechanismen des menschlichen Denkens und ist daher komplex und vielschichtig [11]. Nachdem in diesem Abschnitt zunächst die Kommunikation zwischen Mensch und Computer betrachtet wird, folgt danach ein Exkurs in das Themenfeld der Farbwirkung und Farbpsychologie. Zum Abschluss wird der Begriff intuitives Design auf der Grundlage der psychologischen Erkenntnisse zur menschlichen Intuition näher betrachtet. In dieser Arbeit wird der Fokus auf die menschliche Psychologie bei der Wahrnehmung von Design gerichtet. Die Physiologie des Menschen, die unter anderem durch die Sinnesorgane beeinflusst wird, wirkt sich zwar auch entscheidend auf die Informationsverarbeitung des Menschen aus, ihre nähere Erläuterung würde jedoch den Rahmen der Arbeit überschreiten.

2.4.1 Kommunikation Mensch und Computer

Unser modernes Leben ist geprägt von der häufigen Interaktion zwischen Mensch und Maschine. Ziel dieser Interaktion ist das gemeinsame Bewältigen verschiedener Aufgaben, wie beispielsweise das Bearbeiten eines Fotos. Trotz ihrer Gemeinsamkeiten in der Informationsverarbeitung unterscheiden sich Mensch und Maschine jedoch in vielerlei Hinsicht [6].

Der Mensch ist, vergleichbar mit dem Computer, ein informationsverarbeitendes System. Im Gegensatz zu technischen Systemen wirken auf die Informationsverarbeitung beim Menschen zusätzlich körperliche und geistige Faktoren ein [12]. Dadurch ist die zwischenmenschliche Kommunikation deutlich vielschichtiger, als die zwischen Mensch und Maschine. Der Mensch verfügt über „natürliche Intelligenz und ein weit verzweigtes Weltwissen“ [6], Computer hingegen besitzen lediglich spezielles Wissen und sie sind nicht in der Lage von ihren Algorithmen abzuweichen. Menschen sind dadurch in ihrer Kommunikation flexibler und können Lücken im Kontext durch ihre Erfahrungen interpolieren (ebd.).

In der Einleitung wird die Vermutung aufgestellt, dass zwischenmenschliche Kommunikation zu einem großen Teil auf einer nicht sichtbaren, einer instinktiven Ebene stattfindet. Mit FREUDS Verhaltenstheorie kann diese These gestützt werden. Er nennt dieses Phänomen „Übertragung“ [8] und stellt fest, dass Patienten der Psychotherapie ungewollt und unbewusst Gefühle auf den Therapeuten übertragen, die eigentlich einem anderen Menschen gelten. Durch das „in den Hintergrund treten“ gibt er seinen Patienten die ungehinderte Möglichkeit, diesen Emotionen freien Lauf zu lassen, um sie dann anschließend in ihrer reinsten Form analysieren zu können. In der Verhaltenstherapie spricht man hierbei von „Resonanz“. Resonanz bedeutet im weitesten Sinne, dass der Mensch sein Erleben und seine Wahrnehmung ständig an bereits angelegte Lebenserfahrungen knüpft. Er versucht permanent neue Situationen auf bereits erlebte zurückzuführen und daraus sein zukünftiges Handeln abzuleiten [11].

Auch RASMUSSEN beschreibt in seinem Modell zur menschlichen Kognition (siehe Abb. 5), dass ein Großteil der Informationsverarbeitung des Menschen unterhalb der Bewusstseinschwelle stattfindet. Unter dieser Schwelle muss eine Handlung nicht bewusst initiiert werden, sondern geschieht automatisiert. Nur in ausgewählten Fällen handeln Menschen ganz bewusst. Dies geschieht beispielsweise beim Auftreten einer Inkongruenz während einer automatisierten Handlung oder der bewussten Entscheidung für die selektive Aufmerksamkeit. Praktisch greifbarer wird diese Theorie an folgender Situation: Ein erfahrener Autofahrer wechselt ohne weitere Überlegung

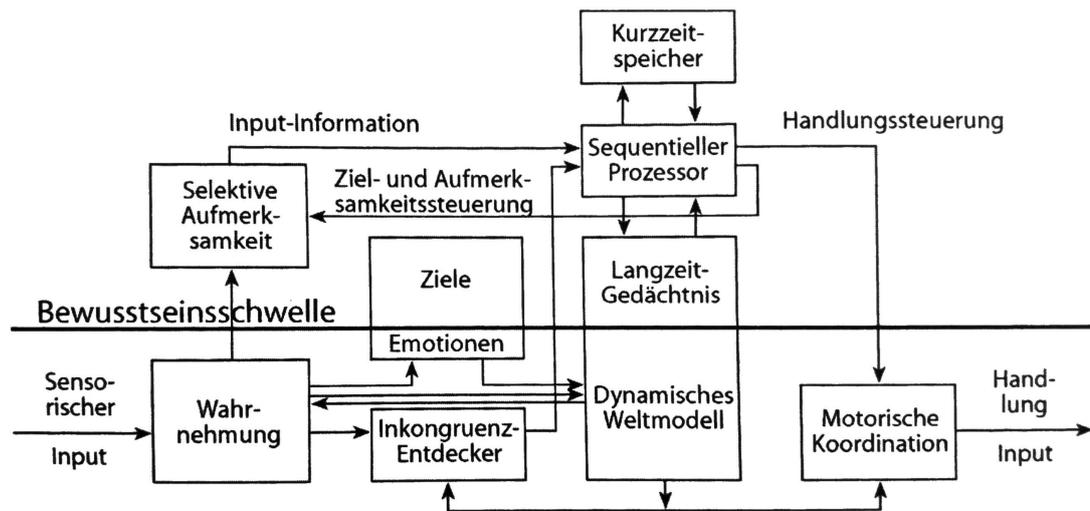


Abbildung 5: Das RASMUSSEN-Modell der menschlichen Kognition (nach HEINECKE) zeigt, wie Informationen verarbeitet werden. Es verdeutlicht den Unterschied zwischen bewussten und unbewussten Handlungen (Abbildung aus [12]).

die Gänge während der Fahrt. Es ist nicht mehr notwendig, dass er darüber aktiv nachdenkt. Passiert jedoch etwas Unerwartetes wie das Klemmen eines Ganges, wird eine Inkongruenz wahrgenommen und führt zum Übergang in den bewussten Teil der Wahrnehmung [12].

Diese Theorien sind für die vorliegende Arbeit relevant, weil sie Aufschluss darüber geben, wie intuitive Software gestaltet werden sollte. Schafft es ein Interface, Mechanismen anzusprechen, die unterhalb der menschlichen Bewusstseinschwelle wirken, kann es intuitiv, also ohne bewusste Handlungsentscheidung, bedient werden.

Damit Software dieses Ziel erreichen kann, muss sie nutzerzentriert gestaltet werden und das „tun, was der Anwender braucht“ [6]. Dazu gehören unter anderem eine Anwenderanalyse im Voraus und eine Gestaltung des Programms gemäß menschlicher Kommunikationsfähigkeiten (ebd.).

2.4.2 Farbwirkung und Farbpsychologie

Im Gegensatz zu Assoziationen mit Formen oder Mustern, sind Farbassoziationen bereits ohne individuelle, eigene Erfahrung wirksam. Farbe wird noch vor Formen und Mustern erkannt und ist daher entscheidend für den ersten Eindruck des Designs. Bei Farbassoziationen werden Farben unterbewusst bestimmte Eigenschaften zugeschrieben. Durch den bewussten Einsatz von Farben im Design können die Assoziationen des Betrachters gesteuert werden und beispielsweise eine positive Attribuierung hervorrufen. Die Farbe grün hat beispielsweise eine beruhigende Wirkung,

weil sie häufig mit der Natur in Verbindung gebracht wird. Weitere Beispiele für Farbassoziationen sind:

Rot	-	<i>Gefahr</i>
Blau	-	<i>Unendlichkeit/Weite</i>
Gelb	-	<i>Freude/Optimismus</i>
Orange	-	<i>Positivität/Lebensfreude</i>
Grau	-	<i>Neutralität</i>

Die Farbwahrnehmung ist subjektiv und wird von verschiedenen Faktoren, wie unter anderem der kulturellen Prägung beeinflusst und ist damit auch Lernprozessen unterworfen. Daher ist es nur begrenzt möglich, aus den Farbassoziationen Ableitungen bezüglich des Designs eines Interfaces zu treffen [11]. Ob ein Design die gewünschte Wirkung erzielt, kann jedoch zielgruppenbezogen überprüft werden.

Die Art und Weise, wie verschiedene Farben innerhalb eines Designs zusammenspielen und in welchem Verhältnis sie zueinander stehen, nennt man Kontrast. Neben dem Strukturieren des Inhalts ist eine der Hauptaufgaben von Kontrasten in Nutzeroberflächen das Erhöhen der optischen Differenzierbarkeit der einzelnen Objekte, um beispielweise eine gute Lesbarkeit zu gewährleisten (ebd.).

2.4.3 Intuition und intuitives Interface

Auf der Ebene der Kognition wird der Begriff Intuition als „*das unmittelbare, nicht diskursive, nicht auf Reflexion beruhende Erkennen, Erfassen eines Sachverhalts oder eines komplizierten Vorgangs*“ oder als „*[...] plötzliches, ahnendes Erfassen*“⁶ definiert. Ein intuitives Interface ist folglich eine Maschinenschnittstelle, deren Bedienung unmittelbar, ohne Erklärung oder Nachdenken vom Anwender verstanden und benutzt werden kann.

Ein Interface ist unter anderem dann intuitiv, wenn der Nutzer es verwenden kann, ohne sich darüber bewusst zu sein, welche sensorisch-motorischen Aspekte die Schnittstelle voraussetzt [2]. Die Navigation innerhalb der Software geschieht zwar nicht völlig unterhalb der, im Abschnitt 2.4.1 vorgestellten ‚Bewusstseinschwelle, aber es kommen unterbewusste Faktoren zum tragen. Der Erkenntnisprozess basiert auf bereits angelegtem Wissen, das unbewusst zum Einsatz kommt und bietet somit die Grundlage für intuitives Verhalten [10].

Da die Intuition jedoch eng mit der Erfahrungswelt eines Menschen verknüpft ist, fällt es schwer, absolute Aussagen über den Grad der Intuitivität einer solchen Schnittstelle zu treffen.

⁶<https://www.duden.de/rechtschreibung/Intuition>

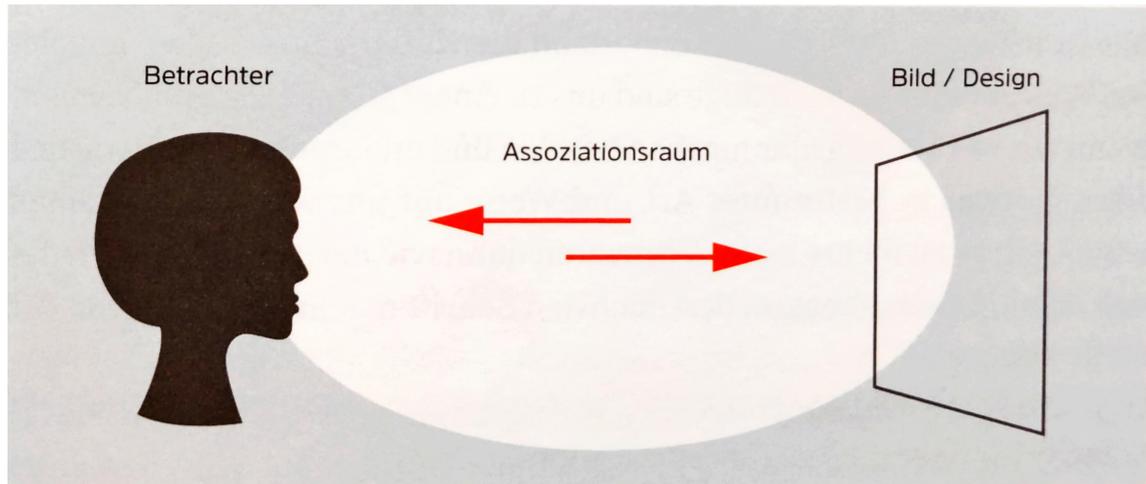


Abbildung 6: Der *Assoziationsraum* zwischen Betrachter und Design sorgt für eine, von Mensch zu Mensch abweichende, Interpretation des Gesehenen (Abbildung aus [11]).

Diesen Umstand verdeutlicht das folgende Beispiel:

Der Ausgangspunkt ist eine Schachpartie. In einer bestimmten Spielsituation trifft ein Mensch in seiner ersten und in seiner fünfhundertsten Partie eine intuitive Entscheidung über den nächsten Spielzug. Die Wahrscheinlichkeit, mit der derselbe Mensch sich in genau der gleichen Situation beide Male für einen anderen Spielzug entscheidet, ist sehr hoch. Grund dafür ist sein gewachsener Erfahrungsschatz.

Zwischen Betrachter und Design öffnet sich ein Assoziationsraum (siehe Abb. 6), der dafür sorgt, dass ein Interface von Nutzer zu Nutzer verschieden wahrgenommen wird [11]. Dadurch unterscheidet sich der Grad der intuitiven Bedienbarkeit je nach Anwender(-gruppe) und sollte nicht verallgemeinert auf alle Nutzergruppen bezogen werden [10]. Um die Entstehung dieser Diskrepanz zu vermeiden, muss eine intuitive Entscheidung auf universellen Erfahrungen basieren, die jeder Anwender vorweisen kann. BAERENTSEN schlägt dazu mittels seiner Analyse der Evolution des menschlichen Gehirns das Zurückgreifen auf das Prinzip der Fortbewegung und der Manipulation von Gegenständen vor. Informationen sollen seiner Meinung nach als Objekte präsentiert werden, die „greifbar“, oder „manipulierbar“ sind [2].

Gestalter von Nutzeroberflächen müssen daher zwei Ausprägungen von Intuition berücksichtigen. Einerseits können UI-Elemente Intuition beim Anwender auslösen, indem sie auf sein bereits erworbenes Wissen oder seine Erfahrungen zurückgreifen. Der Nutzer erfasst intuitiv die Bedeutung und Funktion der Bedienelemente und kennt somit die Handlungsmöglichkeiten, die ihm das Programm bietet. Andererseits können Wissenslücken auch durch unbemerktes Lernen als Produkt der Nutzung geschlossen werden [10]. Verändert sich beispielsweise das Symbol des Mauszeigers beim Überfahren eines UI-Elementes, kann der Nutzer Funktionen daraus ableiten.

2.5 Mathematischer Hintergrund

Dieser Abschnitt gibt einen Überblick über die wichtigsten mathematischen Grundlagen im Kontext der graphbasierten UI-Entwicklung. Nach einem Einstieg in die Graphentheorie folgt eine kompakte Einführung in die Bézierkurven.

2.5.1 Graphentheorie

Die Graphentheorie ist ein, für die Informatik bedeutsames, Teilgebiet der Diskreten Mathematik. Graphen sind vielseitig einsetzbar und dienen unter anderem der Modellierung von Beziehungen oder hierarchischen Strukturen. Das Flussdiagramm, als Sonderform des Graphen, veranschaulicht nacheinander auftretende Programmschritte eines Algorithmus. Ein Graph

$$G = (E, K)$$

ist eine auf einer (endlichen) Menge von Ecken E und einer Menge von Kanten K

$$K \subseteq \binom{E}{2}$$

der Paare $\{u, v\}, u \neq v$ erzeugte binäre Relation, bei der zwei Elemente entweder miteinander in Beziehung stehen, oder nicht. Die graphische Darstellung (siehe Abb. 7) des Ecken-Kanten-Systems ähnelt der Darstellung von Funktionen und dient als Hilfsmittel für das Verständnis der abstrakten Struktur des Graphen. Eine für technische Zwecke oftmals geeignetere Darstellungsform bietet die Adjazenzmatrix A

$$A = (a_{ij})$$

mit den Koeffizienten:

$$a_{ij} = \begin{cases} 1 & u_i u_j \in K \\ 0 & \text{sonst.} \end{cases}$$

Sie bietet eine Struktur zur effizienten Verarbeitung und Speicherung von Daten und eignet sich darum besonders für die Anwendung in Computersystemen. Im Falle des ungerichteten Graphen aus Abbildung 7 (links) stellt sich die Adjazenzmatrix durch

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$



Abbildung 7: Links: Ein ungerichteter Graph G mit $E = \{1, 2, 3\}$, $K = \{\{1, 2\}, \{2, 3\}\}$. Rechts: Ein gerichteter Graph G mit $E = \{1, 2, 3\}$, $K = \{(1, 2), (2, 3)\}$

dar. Hervorzuheben ist dabei die Symmetrie der Matrix, sowie die 0'en in der Hauptdiagonalen. Wird das Beispiel in Abbildung 7 (links) durch einen gerichteten Graphen ersetzt wie in Abbildung 7 rechts, verliert die Adjazenzmatrix ihre Symmetrie und stellt sich durch

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

dar. Der Graph besteht nun aus der Eckenmenge E und einer Menge $K \subseteq E^2$ von geordneten Paaren, den gerichteten oder orientierten Kanten [1].

Eine wichtige Rolle in der Graphentheorie im Bereich der Informatik spielt die Betrachtung von Wegen, also zusammenhängenden Kantenfolgen innerhalb von Graphen. Bei einem Graph $G=(E, K)$ mit den beiden Knoten u und v ist der Weg von u nach v eine Folge von benachbarten Knoten u_0, u_1, \dots, u_l mit $u = u_0$ und $v = u_l$ mit der Länge l . Beträgt die Länge eines Weges 0, nennt man ihn *trivialer Weg* [19].

Entsteht dabei ein neuer Graph $G' = (E', K')$, wobei $E' \subseteq E$ und $K' \subseteq K$, heißt dieser *Untergraph* von G [1].

2.5.2 Bézierkurven

Die Bézierkurve ist eine polynomiale Kurve, die über 2-Stütz- und n -Kontrollpunkte definiert wird (siehe Abb. 8). Sie kommt oft dann zum Einsatz, wenn von einer zu zeichnenden Kurve nur Anfangs- und Endpunkt bekannt sind.

Letztere werden über Bernsteinpolynome

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-1}$$

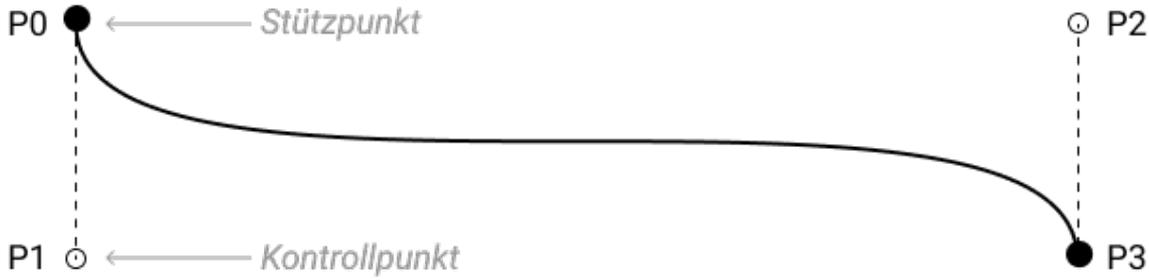


Abbildung 8: Eine Bézierkurve mit den Stützpunkten $\{P0, P3\}$ und den Kontrollpunkten $\{P1, P2\}$.

mit dem Binomialkoeffizient

$$\binom{n}{i} = \frac{n!}{(n-i)!i!}$$

gewichtet, und müssen von der Kurve, im Gegensatz zu den Stützpunkten, nicht durchlaufen werden. Das Bézier-Näherungspolynom ergibt sich zu:

$$P(t) = \sum_{i=0}^n B_i^n(t) P_i$$

Damit die aufwendige Auswertung der Bernsteinpolynome vermieden werden kann, wird häufig der Algorithmus von CASTELJAU zum Approximieren des Polynoms, der sich durch

$$P_i^r(t) = (1-t)P_i^{r-1}(t) + tP_{i+1}^{r-1}(t) \begin{cases} r = 1, \dots, n \\ i = 0, \dots, n-r \end{cases}$$

definiert, verwendet [7, 24].

2.6 Technologie der Umsetzung

Obwohl die Umsetzung einer intuitiven Nutzeroberfläche prinzipiell mit fast allen derzeit verfügbaren Technologien realisierbar ist, eignen sich dennoch manche besser als andere. Um die UI so intuitiv wie möglich zu gestalten, darf der Technologie-Stack an keiner Stelle die Gestaltung des Interfaces restriktieren, sodass der Gestalter seine Ideen frei umsetzen kann.

Weitere Anforderungen an die UI von modernen Applikationen sind beispielsweise Modularität und Erweiterbarkeit, sowie die Lauffähigkeit auf unterschiedlichen Betriebssystemen.

2.6.1 React

*React*⁷ ist eine auf der Scriptsprache *JavaScript* basierende Softwarebibliothek für das Erstellen von interaktiven Weboberflächen. Mithilfe von React lassen sich verkapselte Komponenten erstellen, die ihren eigenen Zustand verwalten können und sich zu komplexen UIs zusammenbauen lassen. Dazu implementiert *React* den *State* einer Komponente. Werden in ihm gespeicherte Variablen aktualisiert, wird die Komponente neu gezeichnet. Die *React-Props* machen es möglich, Daten zwischen verschiedenen Komponenten auszutauschen, ohne das Prinzip der Verkapselung zu verletzen. Weiterhin übernimmt *React* Hintergrundarbeiten wie zum Beispiel das Verwalten des *Data Object Model* (DOM), wodurch die Entwicklung von *React*-Anwendungen vereinfacht wird. *React* unterscheidet zwischen vollwertigen Komponenten und (Stateless-)Funktionskomponenten. Letztere besitzen keinen eigenen *State*, um das zentrale Verwalten von Variablen und somit leichter lesbaren und wartbaren Code zu fördern.

2.6.2 Electron

Durch das Framework *Electron*⁸ schließt sich die Lücke von der Webanwendung zum Betriebssystem. Es ermöglicht das Erstellen von plattformübergreifenden Desktop-Applikationen mithilfe von Webtechnologien wie *JavaScript*, *HTML*, *CSS*. Somit eignet sich auch *React*, um die Nutzeroberfläche einer Desktopanwendung zu erstellen.

⁷<https://www.reactjs.org/>

⁸<https://www.electronjs.org/>

3 VisualArchaeo3D

Ein Teilziel der vorliegenden Arbeit ist die Konzeption und (teilweise) Realisierung einer Nutzeroberfläche für die Software *Archaeo3D*, die den Arbeitstitel *VisualArchaeo3D* trägt. Da eine vollumfängliche Umsetzung der UI den zeitlichen Rahmen dieser Arbeit überschreitet, wird in folgendem Abschnitt ein Zwischenstand des Projektes präsentiert.

Dazu wird zuerst der Nutzungskontext der Anwendung beschrieben. Im Anschluss werden Anforderungen an die UI dargestellt und das Design auf Basis der im Abschnitt 2 gewonnenen Erkenntnisse vorgestellt. Schließlich folgen Erläuterungen zur technischen Umsetzung.

3.1 Beschreibung des Nutzungskontexts

Die zu entwickelnde Nutzeroberfläche baut auf das Softwarepaket *Archaeo3D* auf, das in Abschnitt 2.1.2 vorgestellt wird. Da sich *Archaeo3D* vorrangig an Archäologen richtet, ist die primäre Nutzergruppe bereits gesteckt. Das geplante Interface kommt folglich zu großen Teilen in einem professionellen (Arbeits-)Umfeld zum Einsatz.

Da *Archaeo3D* im Vergleich zu bisherigen Videogrammetrieverfahren einen ressourcenschonenderen Algorithmus implementiert, kann die Nutzung sowohl auf stationären, als auch auf mobilen Endgeräten wie Laptops oder Tablets erfolgen. Auch eine Nutzung im Freien ist damit denkbar. Durch den in Abschnitt 2.6 beschriebenen Technologie-Stack wird, nach Fertigstellung der Software, eine Ausführung unter *Windows*, *Linux* und *MacOS* möglich sein.

Sekundäre Benutzer sind unter anderem 3D-Künstler, Game-Asset-Creators und allgemein 3D-Grafik-Interessierte. Durch die enge Zusammenarbeit mit der Archäologie muss die Annahme getroffen werden, dass der primäre Nutzungskontext ausreichend deutlich überwiegt, sodass sich daraus maßgeblich die Gestaltungsrichtlinien für das Interface ableiten lassen.

3.2 Anforderungen an die UI

Die Anforderungen an die Oberfläche der 3D-Rekonstruktionssoftware ergeben sich vordergründig durch den gegebenen primären Nutzungskontext. Die Rekonstruktionspipeline soll individuell anpassbar sein. Es ist daher notwendig, eine Darstellungsform zu wählen, die sich intuitiv dazu eignet, Anpassungen an der sequenziellen Struktur eines Programms vorzunehmen.

Wie in Abschnitt 2.5.1 erwähnt, eignen sich Graphen unter anderem zur Modellierung von Beziehungen oder hierarchischen Strukturen. Im gegebenen Anwendungsfall ist zu jedem Prozessschritt nur ein nachfolgender Schritt vorgesehen. Deshalb ist von einer 1:1- Beziehung auszugehen.

Die schrittweise Abarbeitung im Fall einer solchen Pipeline kann dabei besonders deutlich durch ein Flussdiagramm, als Sonderform eines gerichteten Graphen, abgebildet werden. Das gedankliche Modell eines Flusses soll beim Nutzer außerdem erste Assoziationen bezüglich der Nutzungsmöglichkeiten der UI wecken. Der (Daten-)Fluss innerhalb des Programms wird über Bézierkurven visualisiert, diese repräsentieren die Kanten des Graphen. Ihre Form wirkt natürlich und erinnert abermals an den Lauf eines Flusses.

3.3 Design der Anwendung

An dieser Stelle wird auf das Design der Oberfläche eingegangen. Nachdem ihre grundlegende Aufteilung erläutert wurde, wird das Konzept der verwendeten Farben und Icons beschrieben. Im Anschluss daran wird der, im Entwicklungsprozess entstandene, Prototyp vorgestellt.

3.3.1 Grundaufteilung

Über den grundlegenden Aufbau der Oberfläche für die spätere Anordnung der Bedienelemente gibt Abbildung 9 Aufschluss. Den Hauptbereich stellt die Fläche für das Verschieben und Neukombinieren der einzelnen Knoten des Prozessgraphen dar. Da der Graph auch im logischen Mittelpunkt der Anwendung steht, soll die Größe des dafür vorgesehenen Containers den Blick des Nutzers lenken und seine Wichtigkeit suggerieren.

Der an den Hauptcontainer unten angefügte Bereich dient der Unterbringung von Steuerelementen zum Starten der Anwendung.

Rechtsseitig ist ein weiterer Container untergebracht, der für die Kapselung allgemeiner Einstellungen der Anwendung und des übergreifenden Projektes zuständig ist. Bei der Skalierung des Anwendungsfensters sollen die dargestellten Proportionen beibehalten werden.

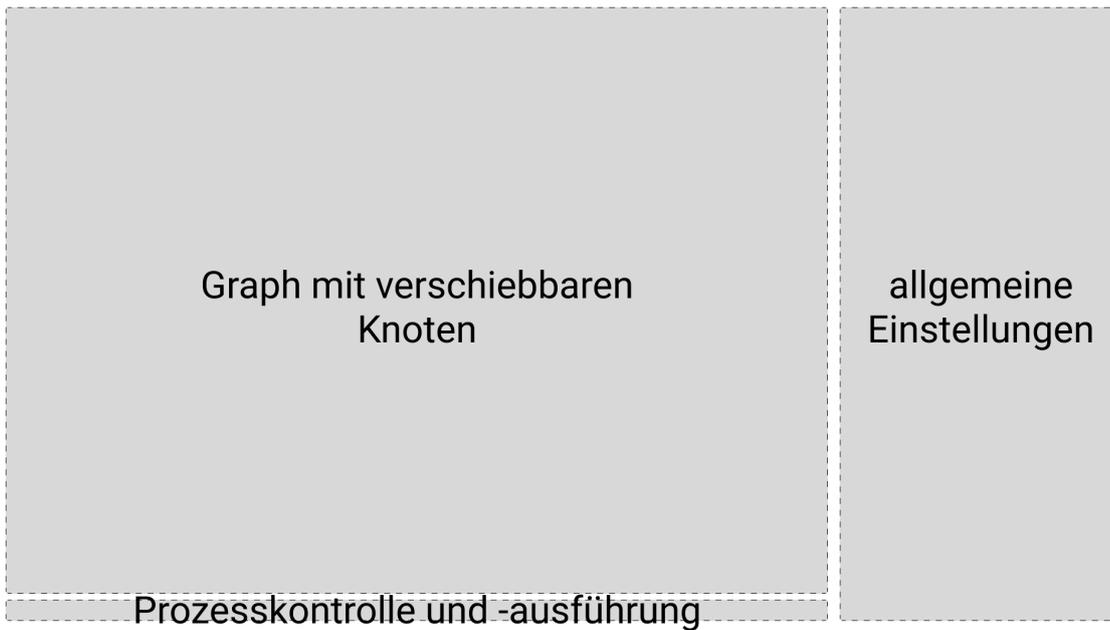


Abbildung 9: Die Nutzeroberfläche teilt sich grundlegend in einen großen Bereich zum Verschieben und Neukombinieren der einzelnen Knoten, einen Container für Grundeinstellungen und einen Bereich für die Elemente zum Ausführen der Prozesskette auf.

3.3.2 Farben, Icons und Typographie

Das Farbschema der Anwendung (siehe Abb. 10, links) leitet sich maßgeblich von den gewünschten Farbassoziationen der, in Abschnitt 3.1 erörterten, primären Nutzergruppe ab. Da die Anwendung unter anderem zu beruflichen Zwecken eingesetzt wird, soll auch die UI Professionalität ausstrahlen. Zu diesem Zweck setzt das Farbkonzept hauptsächlich auf die Farbe grau, die mit den zwei Highlight-Farben dunkelblau und hellgrün in Kontrast steht. Durch den beschriebenen Kontrast zwischen grau und dunkelblau soll das Interface eine beruhigende Wirkung auf den Anwender haben. Gewünschte Assoziationen sind beispielsweise der Himmel am späten Abend oder ein tiefes Gewässer. Die Farbe hellgrün kommt ausschließlich in den Prozessbalken vor und ist eher im Bereich der Pastellfarben angesetzt, um einen geeigneten Kontrast zum gewählten blau zu bieten. Im Alltag ist grün häufig als Symbol für einen erfolgreich beendeten Prozess oder als Startsignal zu finden und bietet dem Anwender daher eine geeignete Verbindung zur Funktion der Prozessbalken.

Die Schriftfarbe ist weiß, um einen guten Kontrast zu den dunklen Primärfarben der Anwendung zu gewährleisten und damit eine gute Lesbarkeit zu erreichen. Als Schriftart wird *Roboto* gewählt, diese wird von *Google*⁹ kostenfrei zur Verfügung gestellt. Sie zeichnet sich durch eine gute Lesbarkeit und in ihrer „light“- Variante

⁹https://fonts.google.com/specimen/Roboto?preview.text_type=custom

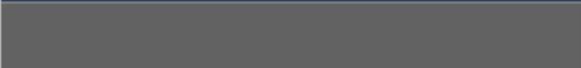
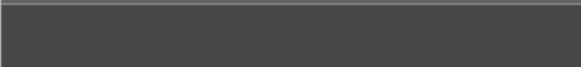
Farben		Icons
	#2E4159 - dunkelblau	
	#626262 - mittelgrau	
	#474747 - dunkelgrau	
	#36FF7A - hellgrün	
	#FFFFFF - weiß	

Abbildung 10: Links: Die in der Nutzeroberfläche hauptsächlich verwendeten, neutralen Grautöne stehen im harmonischen Kontrast zu den Akzentfarben dunkelblau und hellgrün. Schrift und Symbole sind weiß gefärbt. Rechts: Die vier verwendeten Icons fügen sich durch ihre gerundete Form in das bestehende Designkonzept ein.

durch eine filigrane Linienführung aus.

Die unter der *Apache2* Open-Source-Lizenz veröffentlichten Material-Icons von *Google*¹⁰ eignen sich in ihrer abgerundeten Ausführung (siehe Abb. 10, rechts) zur Unterstreichung der zuvor genannten gewünschten Wirkung beim Nutzer. Sie sollen Professionalität und Ruhe, aber auch Modernität vermitteln.

3.3.3 Mockup

Ein mit der Prototypingsoftware *Figma*¹¹ erstelltes Mockup von *VisualArcheo3D*, dient als Leitfaden für die Entwicklung der Nutzeroberfläche. Abbildung 11 zeigt besagtes Mockup in einer, dem Endprodukt nahen, Ausführung. Der Prototyp bildet alle vorkommenden Elemente der entstehenden UI, wie beispielsweise Input-Felder und deren Labels, in einer allgemein gehaltenen Form ab. Durch die Prototyping-Software lassen sich UI-Elemente nahezu mühelos anpassen, ausrichten, hinzufügen oder entfernen.

Hinter den Bedienelementen der UI befindet sich ein quadratisches Liniensystem, das an kariertes Papier erinnert und den Anspruch der Anwendung an Professionalität weiter unterstreichen soll. Bei der Konzeption des Mockups wurde insbesondere darauf geachtet, Steuerlemente und Informationen da anzuordnen, wo der Nutzer sie erwartet. So sind beispielsweise Einstellungen, die ausschließlich einem bestimmten Knoten zugehören, auch in diesem platziert.

¹⁰<https://material.io/resources/icons/?style=round>

¹¹<https://www.figma.com/>

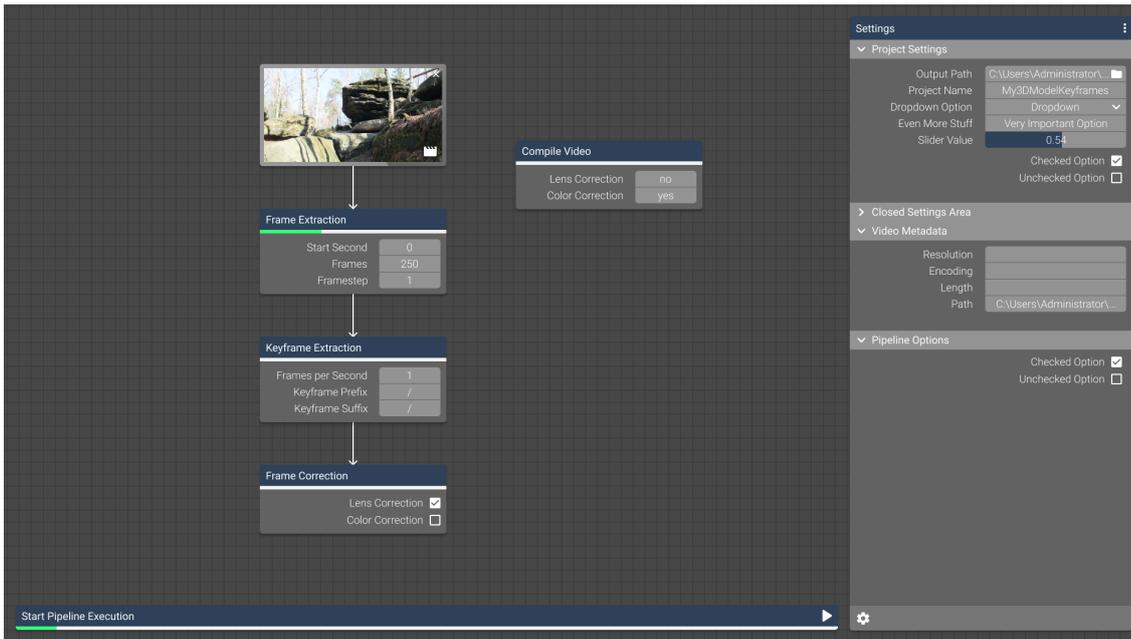


Abbildung 11: Das Mockup zeigt die Nutzeroberfläche in ihrer nahezu finalen Ausführung. Dabei steht der Graph mit seinen verschiebbaren Knoten im Fokus.

3.4 Umsetzung der UI

Mithilfe von Abbildung 12 wird zunächst ein Gesamteindruck der mit *React* und *Electron* erstellten Oberfläche vermittelt. Im Anschluss werden die programmierten Komponenten der UI einzeln vorgestellt.

Zur Umsetzung der Anwendung werden die Node-Package-Manager-Pakete *react-draggable*¹², sowie *react-bezier*¹³ genutzt. Als Ausgangspunkt der *Electron-React-App* dient die, auf der Webseite von *Electronjs* verlinkte, *electron-react-boilerplate*¹⁴.

Trotz der Möglichkeit mit *Electron* eine Crossplattform-App zu erstellen, ist mit derzeitigem Arbeitsstand nur eine Lauffähigkeit auf *Windows* möglich, da die Software im Hintergrund auf *Windows-Batchfiles* zugreift. Um die App auch auf *Linux* ausführen zu können, müssen diese *Batchfiles* ebenfalls für die *Linux-Shell* angeboten werden. Außerdem müssen die verwendeten Javodateien Konsolenvariablen akzeptieren. Erst danach kann im Programmcode eine entsprechende Auswahl getroffen werden.

In der aktuellen Version der UI ist die Hintergrundfunktion der Bézierkurven bereits vorbereitet, jedoch noch nicht vollständig ausprogrammiert. Durch das Klicken eines Ausgangs- und anschließend eines Eingangsankerpunktes wird eine Kurve gezeichnet

¹²<https://www.npmjs.com/package/react-draggable>

¹³<https://www.npmjs.com/package/react-bezier>

¹⁴<https://github.com/electron-react-boilerplate/electron-react-boilerplate>

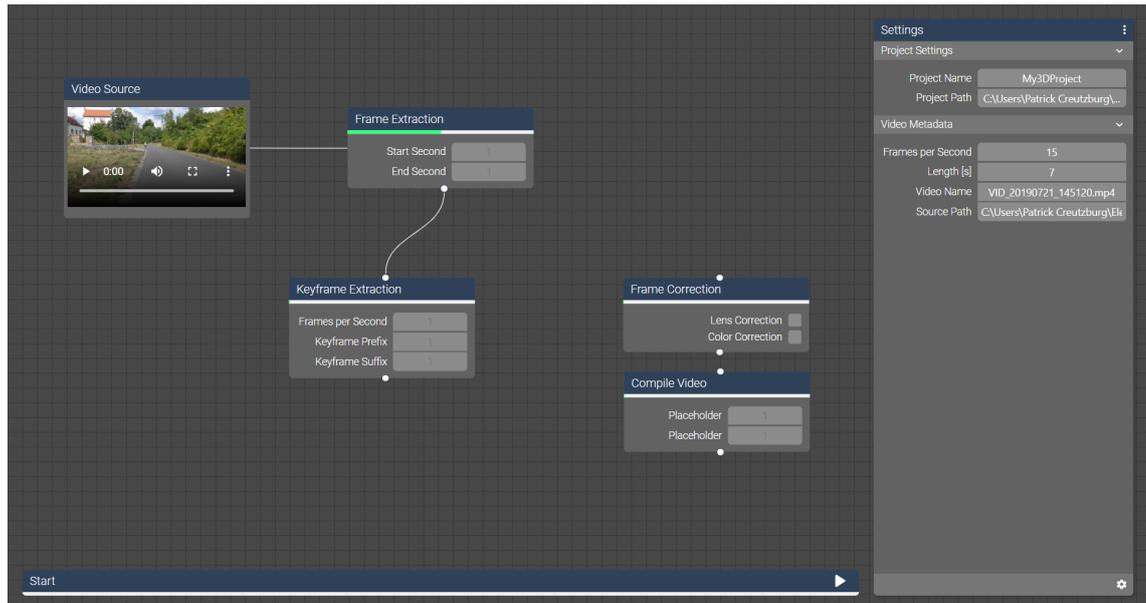


Abbildung 12: Die finale Nutzeroberfläche wird in *React* und *Electron* realisiert. Sie folgt in ihrer Gestaltung dem zuvor präsentierten Mockup.

und die Software erstellt eine Liste mit der Reihenfolge der Knoten. Diese kann verwendet werden, um die Reihenfolge bei der Ausführung der Pipeline zu steuern.

3.4.1 Allgemeiner Prozessknoten

Die Knoten des Prozessgraphen werden durch abgerundete Rechtecke dargestellt. Den allgemeinen Bauplan dazu liefert die Komponente *Node.js*. Jeder Knoten besitzt einen dunkelblau hinterlegten Kopfbereich, mit einer dem Prozessschritt zugehörigen Bezeichnung. Überfährt der Nutzer diesen Bereich mit der Maus, ändert sich der Mauszeiger in ein Pfeilkreuz. Dadurch wird dem Nutzer angezeigt, dass die einzelnen Knoten verschiebbar sind. Hervorzuheben ist dabei, dass jeder Knoten die Einstellungen enthält, die diesem Prozessschritt zugehörig sind. Dadurch kann der Nutzer direkt und intuitiv die Zusammenhänge der UI-Elemente erfassen.

Um eine individuelle Anpassung des Prozessgraphen zu ermöglichen, implementiert die allgemeine Form des Prozessknoten die, über *react-draggable* zur Verfügung gestellte, Drag-and-Drop-Funktion. Durch das Einschließen in den `<Draggable>`-Tag ist der Knoten auf der dafür vorgesehenen Fläche verschiebbar:

```
import Draggable from 'react-draggable';
<Draggable
  bounds="parent"
  handle=".top-bar"
  defaultPosition={{ x: props.x, y: props.y }}

```

```

    >
      <div className="node-container" id={props.cid}>
        ...
      </div>
</Draggable>

```

Einige Merkmale der Knoten-Komponente können durch die übergebenen *Props* individualisiert werden. Bekommt sie von der übergeordneten Komponente beispielsweise eine *inputId*, wird automatisch ein Eingangs-Ankerpunkt angezeigt:

```

{props.inputId ? (
  <div
    className="input-circle"
    onMouseUp={() => props.mouseUp(props.inputId)} //Methode
    der Elternkomponente
    id={props.inputId}
  ></div>
) : null}

```

Das Anzeigen des Ausgangs-Ankerpunkts, sowie des Prozessbalkens wird analog geregelt. Führt der Nutzer das *onMouseDown*-Event auf einem Ausgangs-Ankerpunkt aus, wird die *outputId* des rufenden Knotens an die Funktion *handleCircleButtonDown*

```

handleCircleButtonDown = (outputId) => {
  drawBezierFrom = outputId;
  let newOrder = this.state.executionOrder;
  newOrder.push(drawBezierFrom);
  this.setState({ executionOrder: newOrder });
};

```

übergeben und der entsprechende Knoten wird in die *executionOrder*-Liste eingetragen. Registriert das Programm ein *onMouseUp*-Event über einem Eingangs-Ankerpunkt, führt es die Funktion *handleCircleButtonUp*

```

handleCircleButtonUp = (inputId) => {
  let newOrder = this.state.executionOrder;
  newOrder.push(inputId);
  this.setState({ executionOrder: newOrder });
  this.setState((prevState) => ({
    bezierSettings: [
      ...
    ],
  }));
};

```

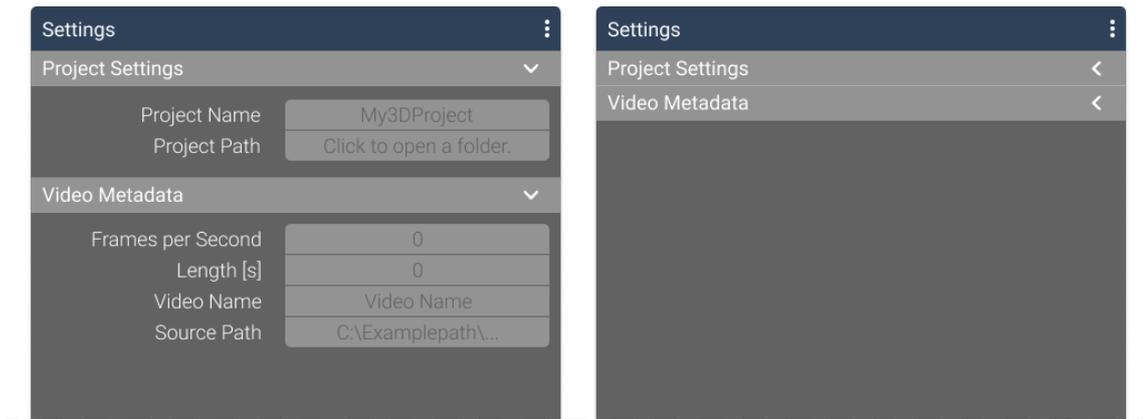


Abbildung 13: Links: Alle verfügbaren Informationen des Settings-Containers werden angezeigt. Rechts: Durch das Einklappen der Collapsible-Komponenten können nicht benötigte Informationen und Einstellungen ausgeblendet werden.

aus. Diese Funktion trägt dann einen weiteren Punkt in die *executionOrder*-Liste ein und legt schließlich die neue Bézierkurve an (siehe Abb. 15 rechts). Der anzuzeigende Inhalt wird dem jeweiligen Knoten wie folgt übergeben:

```
<div className="node - contend">
  {props.children}
</div>
```

Mithilfe des *Props opacityApplied=1* kann der Knoten semitransparent dargestellt werden. Dadurch wird dem Nutzer signalisiert, dass dieser Knoten aktuell nicht beachtet werden soll.

3.4.2 Settings-Container

Die Komponente *SettingsContainer.js* ist verantwortlich für die Kapselung von projektübergreifenden Optionen und das Anzeigen der Video-Metadaten. Zur Anlehnung an das Gestaltungsprinzip der Wiederholung hat sie die selben optischen und strukturellen Merkmale wie die Prozessknoten. Damit größere Mengen von Daten sinnvoll gruppiert werden können, besitzt die Komponente außerdem zusammenfaltbare Elemente, um bei Bedarf Informationen von untergeordneter Wichtigkeit auszublenden (siehe Abb. 13).

Im Hinblick auf die spätere Erweiterung der Anwendung ist die dafür zuständige Komponente *Collapsible.js* von Nutzen. Bei der Erzeugung einer *Collapsible*-Instanz innerhalb der Settings-Komponente lässt sich über die Variable *initialState*

```
<Collapsible title="Project Settings" initialState="1">
```

der Startzustand mit $1 \hat{=} \text{ausgeklappt}$ bestimmen. Um das Ein- und Ausklappen der Komponente zu ermöglichen, verwendet sie den *React-Hook* `useEffect`:

```
const [active, setActive] = useState(props.initialState);
  const contentRef = useRef(null);
  useEffect(() => {
    contentRef.current.style.maxHeight = active
      ? `${contentRef.current.scrollHeight}px`
      : '0px';
  }, [contentRef, active]);
```

Wird die Funktion `toggleActive`

```
const toggleActive = () => {
  setActive(!active);
};
```

aufgerufen, zeichnet sich die Komponente mit der Größe des anzuzeigenden Inhalts erneut. Fährt der Nutzer mit der Maus über den Kopfbereich der *Collapsible*-Komponente, färbt sich ihr Hintergrund heller, um dem Anwender zu zeigen, dass er anklickbar ist.

Das Auf- und Zuklappen der Container wird durch eine 90°-Drehung des beigefügten Pfeilsymbols optisch unterstützt. Beide Bewegungen werden über die CSS-Animationskurve *Ease* umgesetzt, um für den Nutzer optisch ansprechend zu sein.

Durch die Eingabe des Projektnamens in der Project-Settings-Kategorie des Settings-Containers, findet bei Ausführung der Pipeline die Benennung des Projektordners statt. Dessen Zielpfad kann der Nutzer durch einen Mausklick in das Project-Path-Textfeld über einen Systemdialog festlegen.

Fügt der Anwender eine Videodatei in den dafür vorgesehenen Knoten ein, füllt sich der Reiter Video-Metadaten automatisch mit den relevanten Metadaten des Videos.

3.4.3 Die Hauptkomponente

Die Komponente *Main.js* stellt das Herzstück der Anwendung dar. Sie wird beim Start der Software geladen und erzeugt die in Abschnitt 3.3.1 beschriebene Grundaufteilung der UI. Sie verwaltet den gesamten Datenfluss, um diesen innerhalb der Anwendung zu zentralisieren. Der ihr zugehörige *State* speichert alle veränderlichen Daten, die bei Bedarf auch an Unterkomponenten weitergegeben werden können. Die Hauptkomponente verwaltet außerdem die Funktionen für die Ausführung der *Archaeo3D-Javadateien*.

Eine der Besonderheiten von *Electron* ist, dass in Kombination mit *Node.js* der Zugriff auf vollständige Systempfade möglich ist. Dies ist sowohl bei der Weitergabe der Daten an das *Archaeo3D*-Softwarepaket, als auch zum Herausfinden der Video-Metadaten notwendig.

Durch die Weiterleitung der Funktionen für das Zeichnen und Registrieren der Bézierkurven an die Knoten-Komponenten erfährt die Hauptkomponente die IDs selbiger und kann so die Reihenfolge der Pipeline registrieren und im *State* ablegen. Sie erzeugt alle spezialisierten Instanzen der Knoten-Komponente, sowie des Settings-Containers. Diese werden im Folgenden erläutert.

3.4.4 Quellknoten

Der Quellknoten (siehe Abb. 14) steht in jeder möglichen Konfiguration an erster Stelle im Prozessgraphen und ist eine spezialisierte Instanz der Komponente *Node.js*. Er enthält den Hinweis zum Einfügen einer Videoquelle. Zieht der Nutzer eine Videodatei mit der Maus über die *FileDropZone*

```
<FileDropZone
  handleDrop={this.handleDrop}
  getFullPath={this.getFullPath}
>
```

färbt sich sein Hintergrund weiß, um den Abwurfbereich für die Dateien optisch hervorzuheben. Lässt der Anwender das Video über dem weiß gefärbten Bereich fallen, lädt das Programm einen *HTML5*-Videoplayer

```
<video
  className="video"
  controls
  onLoadedMetadata={(e) => {
    {this.handleMetadata(file);}
    this.setState({
      videoLength: Math.floor(e.target.duration),
    });
  }}
>
  <source src={file} />
</video>
```

zur Vorschau der Datei und registriert im Hintergrund ihren Pfad im Dateisystem. Zu diesem Zweck wird die Funktion *handleDrop()*

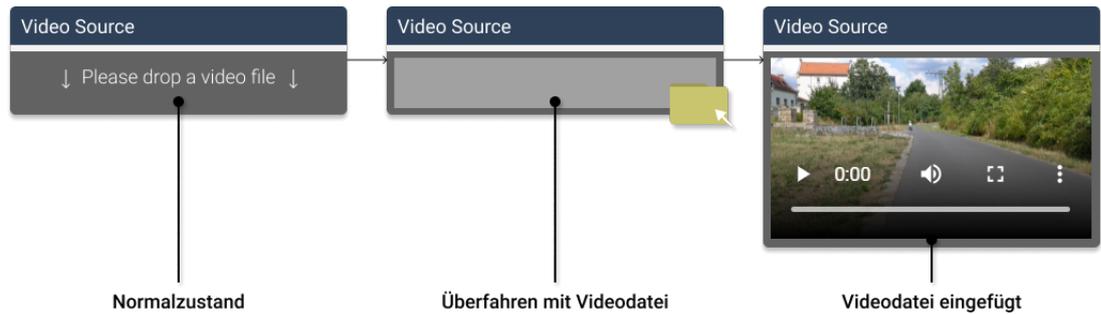


Abbildung 14: Videos sollen vom Nutzer per Drag-and-Drop in den Quellknoten eingefügt werden. Eine entsprechende Aufforderung gepaart mit Signalpfeilen und die farbliche Veränderung des Hintergrundes sollen ihn auf diese Programmfunktion hinweisen.

```

handleDrop = (file) => {
  let fileList = this.state.file;
  for (var i = 0; i < file.length; i++) {
    if (!file[i].name) return;
    fileList.push(URL.createObjectURL(file[0]));
  }
  this.setState({ fileName: file[0].name });
  this.setState({ readOnly: false });
};

```

von der Hauptkomponente *Main.js* an die Komponente *FileDropZone.js* weitergegeben. Diese arbeitet mit einer Liste von Dateien und ist damit für das spätere Arbeiten mit mehreren Quelldateien bereits vorbereitet. Die Komponente *FileDropZone* erfragt dann über

```

this.setState({ filePath: e.dataTransfer.files[0].path });

```

den vollständigen Pfad der Videodatei, schreibt ihn in ihren eigenen *State* und gibt ihn anschließend über

```

this.props.getFullPath(this.state.filePath);

```

an die Hauptkomponente zurück.

Die Frame-Extraction in *Archaeo3D* erwartet die Startsekunde und eine Anzahl von zu extrahierenden Frames ab dem angegebenen Zeitpunkt als Übergabewerte. Damit die Eingabe für den Nutzer intuitiver ist, werden Start- und Endzeitpunkt gleichermaßen in Sekunden eingegeben. Die Nutzereingabe muss dafür zunächst in eine Anzahl von Frames umgerechnet werden. Dazu werden die Länge der gewählten Videosequenz sowie die Bilder pro Sekunde (FPS) des Videos benötigt. Da der verwendete Technologie-Stack die Ausgabe der FPS nativ nicht unterstützt, ist es

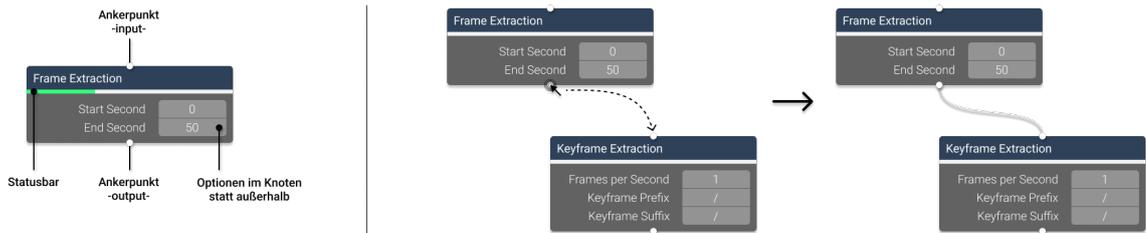


Abbildung 15: Links: Der Frame-Extraction-Knoten besitzt zusätzliche Textfelder zur Eingabe der Start- und Endsekunde der Bildextraktion. Rechts: Führt der Nutzer eine Ziehbewegung von Ausgangs- zu Eingangsankerpunkt zweier Knoten aus, entsteht eine neue Beziér-Verbindungslineie.

notwendig, ein externes Programm zu verwenden. Die Funktion *handleMetadata()* wird beim Einfügen eines Videos per Drag-and-Drop getriggert und übergibt den Videopfad an das *Windows-Batchscript getFPS.bat*:

```
cd C:/../ffmpeg/bin ffmpeg.exe -v error -select_streams v -of
default=noprint_wrappers=1:nokey=1 -show_entries stream=
r_frame_rate %1
```

Das Skript leitet diesen Pfad an das externe Programm „FFmpeg“¹⁵ weiter, das im Anschluss die gewünschte Information über *stdout* zurückgibt. Da die Ausgabe inklusive aller Befehle und Pfade erfolgt, muss die eigentliche Information mittels String-Manipulation extrahiert und anschließend berechnet werden.

3.4.5 Frame-Extraction

Der Frame-Extraction-Knoten (siehe Abb. 15) stellt eine weitere spezialisierte Instanz der Komponente *Node.js* dar und besitzt, neben den allgemeinen Elementen der *Node.js*-Komponente, auch Textfelder zur Eingabe der gewünschten Start- und Endsekunde der Bildextraktion.

Im Folgenden wird exemplarisch die Instanziierung eines Knotens innerhalb der Hauptkomponente ausführlich beschrieben. Da das Vorgehen bei den weiteren Knoten analog ist, werden im späteren Verlauf lediglich ihre Besonderheiten hervorgehoben. Über den Import

```
import Node from '../components/Node';
```

steht der in Abschnitt 3.4.1 beschriebene, allgemeine Prozessknoten der Hauptanwendung zur Verfügung. Danach kann der Knoten mittels des *<Node>*-Tags

¹⁵<https://www.ffmpeg.org/>

```

<Node
  title="Frame Extraction"
  opacityApplied={this.state.fileName == '' ? '1' : '0'}
  draggable="true"
  process={this.state.processOfFrameExtraction}
  xPos={this.state.xPosFrameExtraction}
  yPos={this.state.yPosFrameExtraction}
  containerId="n2"
  outputId="n2o"
  mouseDown={this.handleCircleButtonDown}
  mouseUp={this.handleCircleButtonUp}
>

```

erzeugt werden. Über ihn bekommt der Knoten zur Frame-Extraction, die für ihn notwendigen *Props* übergeben. Dazu gehören der Titel, die IDs des Ausgangs- und Eingangsankerpunkts, die Position, der Fortschritt des Prozessbalkens, die Sichtbarkeit, sowie Mouse-Events die zum Zeichnen der Bézierlinien verwendet werden. Außerdem wird festgelegt, ob der Knoten verschiebbar sein soll. Es folgt der eigentliche Inhalt der Komponente.

Anschließend werden die zu extrahierenden Frames, je nach Reihenfolge der Nutzereingabe über

```

setStartSecond = (event) => {
  this.setState({ startSecond: event.target.value });
  let frames =
    (this.state.endSecond - event.target.value) * this.state.
      videoFPS;
  this.setState({ framesToExtract: frames });
};

```

oder bei Ersteingabe der Endsekunde analog über

```

let frames =
  (event.target.value - this.state.startSecond) * this.state.
    videoFPS;

```

errechnet.

3.4.6 Ausführungs- und Kontrolleinheit

Die Komponente *Executer.js* (siehe Abb. 16) dient der Ausführung der Videogrammetrie-Pipeline. Sie enthält, neben einer Play-Button-Grafik zum Symbolisieren des

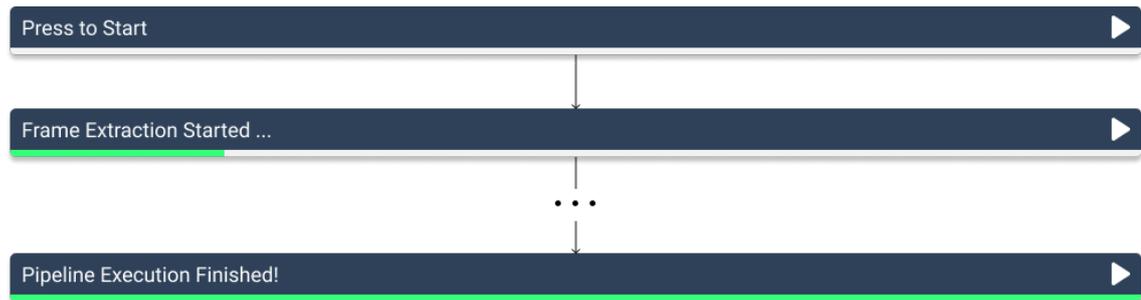


Abbildung 16: Die Komponente `Executer.js` führt bei einem Mausklick die (individuelle) Pipeline aus und gibt Auskunft über den aktuellen Status der Anwendung.

Ausführens der Pipeline, einen Prozessbalken, der zur Anzeige des Gesamtfortschritts genutzt wird. Sie stellt das Hauptinstrument zum Vermitteln des aktuellen Zustandes der Anwendung dar und gibt eine dazu passende Nachricht aus.

Um die Reihenfolge der einzelnen Prozessknoten individualisierbar zu machen, ist es notwendig, die den Knoten zugehörigen Hintergrundprozesse voneinander zu isolieren. Dazu besitzt die Hauptkomponente je Knoten eine ausführende Funktion, die je nach Konfiguration der Pipeline zentralisiert über die Funktion `executePipeline` aufgerufen werden kann.

Nachdem `executePipeline` überprüft hat, ob alle notwendigen Nutzereingaben vorhanden sind, legt sie den Projektordner am gewünschten Zielpfad über

```
let projectFolder = this.state.projectPath + '\\\\' + this.state.projectName;
if (!fs.existsSync(projectFolder)) {
  fs.mkdirSync(projectFolder, (err) => {
    if (err) throw err;
  });
}
```

an. Dieser Vorgang geschieht synchron, da der Projektordner für den weiteren Verlauf zwingend angelegt sein muss. Im Anschluss folgen die jeweiligen Funktionsaufrufe der einzelnen Prozessknoten, die wie folgt aufgebaut sind:

1. Konsolen-Logging relevanter Daten
2. Anlegen des eigenen Ordners innerhalb des Projektordners
3. Synchrones Ausführen der zugehörigen Javodateien

Das zum Ausführen der *Archaeo3D*-Javodateien verwendete Vorgehen zeigt die der Frame-Extraction zugehörige Funktion `executeFrameExtraction`, die über

```

const execSync = require('child_process').execSync;
execSync(
  "C../../Java/ExtractVideo.bat"
  "${this.state.fullPath}"
  "${fileExtractionFolder}"
  "${fileExtractionFolder+'\\logfile.txt'}"
  "${this.state.startSecond}"
  "${this.state.framesToExtract}"
  "${this.state.framestep}"',
  (err, stdout, stderr) => {
    if (err) {
      console.error(err);
      return;
    }
    console.log(stdout);
  }
);

```

die zugehörige *Windows-Batchdatei ExtractVideo.bat*

```

cd C../../Java java ExtractVideo_console "C:\Program Files(x86)\
MPlayer_forWindows\mplayer.exe" %1 %2 %3 %4 %5 %6

```

aufruft, die wiederum eine zuvor vorbereitete Konsolenversion eines *Archaeo3D*-Teilprogrammes startet. Eine synchrone Ausführung der einzelnen Prozessschritte ist notwendig, da diese teilweise aufeinander aufbauen. Weiterhin wird somit ermöglicht, jeweils nach dem Beenden eines Prozessschrittes den Fortschritt der entsprechenden Prozessbalken zu aktualisieren.

3.4.7 VisualArchaeo3D-Testdurchlauf

Bevor die entwickelte UI einem Usability-Test unterzogen werden kann, muss sie zunächst auf ihre vollständige Funktionalität geprüft werden. Dazu finden mehrere Testläufe mit verschiedenen Quelldateien und Eingabewerten statt. Die Abbildungen 17 und 18 zeigen einen solchen Durchlauf.

Damit der Test einem späteren Anwendungsszenario so gut wie möglich entspricht, dient als Videomaterial die Unterwasseraufnahme eines Sees. Der obere Teil von Abbildung 17 stellt den Ausgangspunkt des Tests dar. Zu sehen ist das Programm im Startzustand, der geöffnete Ordner als Testumgebung sowie die zum Abgleich von *Windows* ausgelesenen Metadaten des Videos.

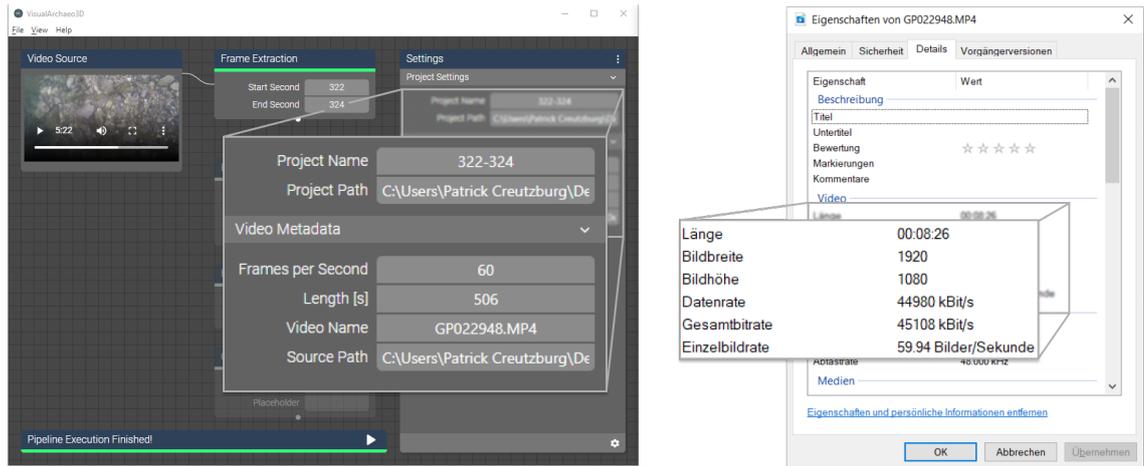


Abbildung 17: Die Metadaten des Videos stimmen auf Ebene des Betriebssystems mit denen von *VisualArcheo3D* überein.

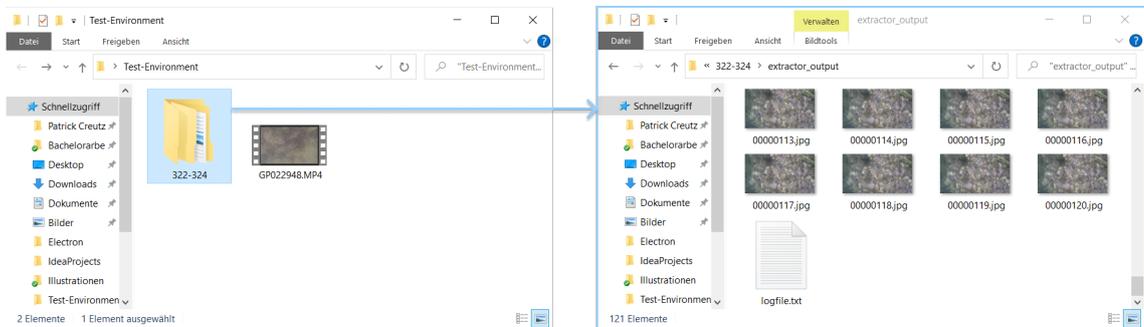


Abbildung 18: Die Ausführung der Pipeline erzeugt den Zielordner in der Testumgebung und extrahiert die gewünschte Anzahl von Bildern.

Den Zustand nach Ausführung der *Keyframe-Extraction* zeigt der untere Teil der Abbildung. Dabei ist zunächst festzustellen, dass die von *VisualArcheo3D* ausgelesenen Metadaten denen der *Windows-Eigenschaften* gleichen. Wie durch

$$\Delta t = \text{EndSecond} - \text{StartSecond}$$

und

$$\text{FramesToExtract} = \Delta t \cdot \text{FramesPerSecond}$$

nachzurechnen, wurden 120 Bilder des Videomaterials, in den dafür vorgesehenen Ordner, extrahiert.

Die Anwendung reagiert sofort auf Nutzereingaben, Performance-Probleme sind nicht festzustellen.

4 Usability-Test

Den Kern dieses Kapitels bildet ein *Remote-Usability-Test*, der in Form eines leitfadenbasierten Interviews durchgeführt wird. Im folgenden Abschnitt werden die Erhebungsinstrumente, sowie die Zielsetzungen und Rahmenbedingungen des Tests näher vorgestellt. Anschließend werden die Ergebnisse der Untersuchung analysiert und präsentiert. Auf Basis der gesammelten Erkenntnisse folgen Verbesserungsvorschläge zur Optimierung von *VisualArcheo3D*.

4.1 Zielsetzung

Der Usability-Test soll Aufschluss darüber geben, ob die in den vorherigen Abschnitten angeführten theoretischen Grundlagen erfolgreich umgesetzt wurden und eine intuitive Bedienbarkeit von *VisualArcheo3D* möglich ist. Weiterhin soll der Test Erkenntnisse hinsichtlich der optischen Gestaltung der Anwendung liefern. Die Befragung soll als Grundlage für weitere Entwicklungsschritte dienen, indem sie positive Aspekte hervorhebt und etwaige Probleme aufzeigt.

4.2 Methode

In diesem Abschnitt werden die Erhebungsinstrumente und die Rahmenbedingungen des Tests dargestellt. Der besondere Fokus liegt dabei auf der *Thinking-Aloud-Methode* sowie der Vorstellung der erhobenen Usability-Metriken. Zunächst werden dazu die Probanden beschrieben, danach folgt eine Begründung des Studiendesigns. Außerdem wird erläutert, unter welchen weiteren Rahmenbedingungen der Usability-Test durchgeführt wird.

4.2.1 Probandenvorstellung

Der Test wird mit sechs Probanden durchgeführt. Eine detaillierte Zielgruppenanalyse fand im Vorfeld der Erhebung aufgrund der begrenzten Zeit nicht statt. Dennoch ähneln sich die Befragten in Alter und ihren technischen Vorkenntnissen der in Abschnitt 3.1 beschriebenen Zielgruppe von *VisualArcheo3D*. Der Usability-Test kann außerdem in ähnlicher Form im weiteren Software-Entwicklungsprozess erneut durchgeführt werden, um eine kontinuierliche Qualitätssteigerung zu erreichen. Im Vorfeld von späteren Befragungen sollte dann eine genaue Nutzeranalyse stattfinden. Der im Rahmen dieser Arbeit durchgeführte Usability-Test dient dennoch zur Exploration des Forschungsfeldes.

Es haben drei männliche und drei weibliche Probandinnen am Test teilgenommen. Diese waren im Alter von 28 bis 52. Fünf Probanden besitzen einen Hochschulabschluss, eine Probandin hat eine Ausbildung absolviert. Zwei Personen gehen einem technischen Beruf nach. Die anderen Probanden arbeiten im Bildungs- und Dienstleistungssektor. Alle Teilnehmer nutzen in ihrem beruflichen oder privaten Umfeld technische Geräte, hauptsächlich verwenden sie Desktop-PCs, Laptops, Smartphones und Tablets. Sie sind im Umgang mit gängigen Textverarbeitungsprogrammen vertraut und nutzen individuell weitere Software auf Basis ihres beruflichen Hintergrunds. Die allgemeine Sicherheit im Umgang mit Software schätzten jeweils zwei Probanden als „sehr sicher“, zwei als „sicher“ und zwei als „weniger sicher“ ein.

NIELSEN stellt fest, dass drei bis fünf Probanden ausreichen, um aussagekräftige Ergebnisse bei Usability-Tests zu erhalten. Da die getesteten Personen dieser Erhebung jedoch nicht eindeutig dem konkreten Anwenderkreis zuzuordnen sind, ermöglicht die Befragung weiterer Probanden eventuell einen größeren Erkenntniszuwachs [21]. Dennoch ist auch dieser Usability-Test relevant, da gerade die Thinking-Aloud-Methode gut dazu geeignet ist, in einem frühen Stadium der Softwareentwicklung erste Erkenntnisse zu sammeln. Diese werden später dazu genutzt, um die Software noch besser auf den Anwender abzustimmen [13].

4.2.2 Begründung des Studiendesigns

Der Usability-Test basiert hauptsächlich auf der *Thinking-Aloud-Methode* und der Erhebung von quantitativen Usability-Metriken.

Bei der *Thinking-Aloud-Methode* werden Probanden spezifische Aufgaben (Use-Cases) gestellt, die so realitätsnah wie möglich gestaltet sind. Alle Einfälle und Gedanken, die den Anwendern bei der Bewältigung dieser Aufgaben in den Sinn kommen, sollen während des Tests laut ausgesprochen werden. Währenddessen werden die Äußerungen der Teilnehmer protokolliert, der Moderator darf die Aufgabenstellung dabei inhaltlich nicht erläutern oder das Handeln der Probanden kommentieren [13]. Ein solcher Usability-Test läuft nach einem stets gleichbleibenden Schema ab:

1. Begrüßung und Briefing
2. Aufgabenstellung
3. Durchführung der Aufgabe
4. Optionales (standardisiertes) Interview
5. Abschluss und Verabschiedung

Das Briefing zu Beginn des Tests ist besonders wichtig, da die Probanden darin bestärkt werden müssen, tatsächlich alle Gedanken während der Aufgabenbearbeitung zu verbalisieren. Den Testern muss bewusst werden, dass ihre ausgesprochenen Gedanken und Absichten genauso relevant sind, wie das Bearbeiten der Aufgaben selbst. Außerdem ist, insbesondere bei dieser Methode darauf zu achten, dass Moderator und Protokollant möglichst wenig auf die Tester einwirken (ebd.).

Nachdem die Tester eine Aufgabe erfolgreich erledigt oder abgebrochen haben, folgt jeweils eine Frage, um herauszufinden, wie Probanden den Schwierigkeitsgrad der zuvor bearbeiteten Aufgabe einschätzen. Für die Beantwortung der Fragen wurde eine Likert-Skala¹⁶ ohne neutrale Antwortmöglichkeit gewählt, um die Tendenz zur Mitte¹⁷, zu vermeiden.

Im Anschluss an die Aufgabenbearbeitung folgen, als Post-Session-Interviews, zwei standardisierte Befragungen, die eine bessere Vergleichbarkeit zwischen der aktuellen Version von *VisualArcheo3D* und späteren Überarbeitungen schaffen sollen. Obwohl diese Befragungen quantitative Items erheben, sind sie bei geringen Stichprobengrößen nicht repräsentativ. In der Praxis ist dies jedoch auch nicht notwendig, da sie bereits in einem kleinen Untersuchungsrahmen wichtige Rückschlüsse auf die Bedienbarkeit und die Gestaltung einer Software geben können [14].

Die erste standardisierte Schlussbefragung bildet die Kurzform des *Visual Aesthetics of Website Inventory Tests (VisAWI-S)*. Dieser stellt vier Fragen zur subjektiven Einschätzung der Ästhetik der Nutzeroberfläche. Die Testperson gibt dabei auf einer Skala von eins bis sieben an, ob und wie stark sie einer Aussage zustimmt. Zu den Stärken des *VisAWI-S* gehören vor allem seine schnelle Durchführbarkeit und die hohe Validität [9].

Da der *VisAWI-S* Test nur optische Aspekte berücksichtigt, kommt zusätzlich der *System Usability Scale* Fragebogen (*SUS*) zur Anwendung. Hierbei soll jeder Proband auf einer Skala von null bis vier beantworten, wie sehr er einer Aussage zustimmt. Dabei sind die Aussagen abwechselnd positiv und negativ formuliert. Auch dieser Test ist schnell durchzuführen und auszuwerten und liefert auch bei kleiner Stichprobengröße bereits verlässliche Tendenzen zur Bewertung der Qualität einer Oberfläche (ebd.).

4.2.3 Durchführung

Der Nutzertest von *VisualArcheo3D* wurde als moderierter Remote-Usability-Test durchgeführt. Dabei befanden sich die Probanden im häuslichen Umfeld. Pro Test-

¹⁶<https://www.dorsch.hogrefe.com/stichwort/likert-skala>

¹⁷<https://www.dorsch.hogrefe.com/stichwort/tendenz-zur-mitte>

sitzung war jeweils ein Testteilnehmer anwesend. Testleiter und Probanden kommunizierten über das Videokonferenz-Tool *jitsi*¹⁸. Zu Beginn der Befragung erfolgte die Erläuterung des Nutzungskontextes der Software und die Testteilnehmer wurden auf die *Thinking-Aloud-Methode* eingestimmt.

Die Durchführung des Usability-Tests wurde durch die Remote-Desktop-Anwendung *TeamViewer*¹⁹ realisiert. Durch den Fernzugriff der Probanden auf die Testumgebung kam es jedoch teilweise zu merklichen Verzögerungen, die bei der Auswertung des Tests berücksichtigt werden müssen.

Im Verlauf der Befragung protokollierte der Testleiter die Angaben der Nutzer und notierte ihre Aussagen während der Aufgabenbewältigung. Jeder Testlauf hat durchschnittlich 30 min in Anspruch genommen.

4.3 Darstellung und Auswertung der erhobenen Daten

Die folgende Auswertung der Aufgaben des Usability-Tests erfolgt in der Reihenfolge ihres Auftretens im Fragebogen. An erster Stelle stehen die Use-Case Aufgaben. Darauf folgen die Fragebögen zur visuellen Wahrnehmung und zur Usability.

Die Namen der Testpersonen entsprechen dem Geschlecht der Teilnehmer, weichen jedoch von den echten Namen zum Zwecke der Anonymisierung der Probanden ab.

4.3.1 Auswertung der Use-Cases

Da die Use-Cases sich, im Gegensatz zu den standardisierten Fragebögen, speziell auf die zu testende Anwendung beziehen, werden sie im folgenden einzeln analysiert. Neben den Antworten auf die Schwierigkeit der gestellten Aufgaben, wird zur Auswertung auch das *Thinking-Aloud* Protokoll herangezogen.

Um die Teilnehmer an den Test zu gewöhnen, sind die Aufgaben in steigender Schwierigkeit gestellt. Die Anschlussbefragung der Probanden wurde pro Use-Case ausgewertet. Abbildung 19 zeigt prozentual, für welche der Antworten im Bereich „sehr einfach“ bis „sehr schwer“ sich die Teilnehmer jeweils entschieden haben.

Fügen Sie das gewünschte Video in *VisualArcheo3D* ein.

Allen Teilnehmern ist es ohne Probleme gelungen, ein Video in die Nutzeroberfläche einzufügen, wobei zwei Probanden betonten, der Beschriftung „*Drop a file here.*“ gefolgt zu sein. Außerdem wurde das Feedback des Quellknotens, in Form der farbigen Hinterlegung beim Einfügen der Datei, als positiv hervorgehoben. Ein

¹⁸<https://www.meet.jit.si/>

¹⁹<https://www.teamviewer.com/de/>

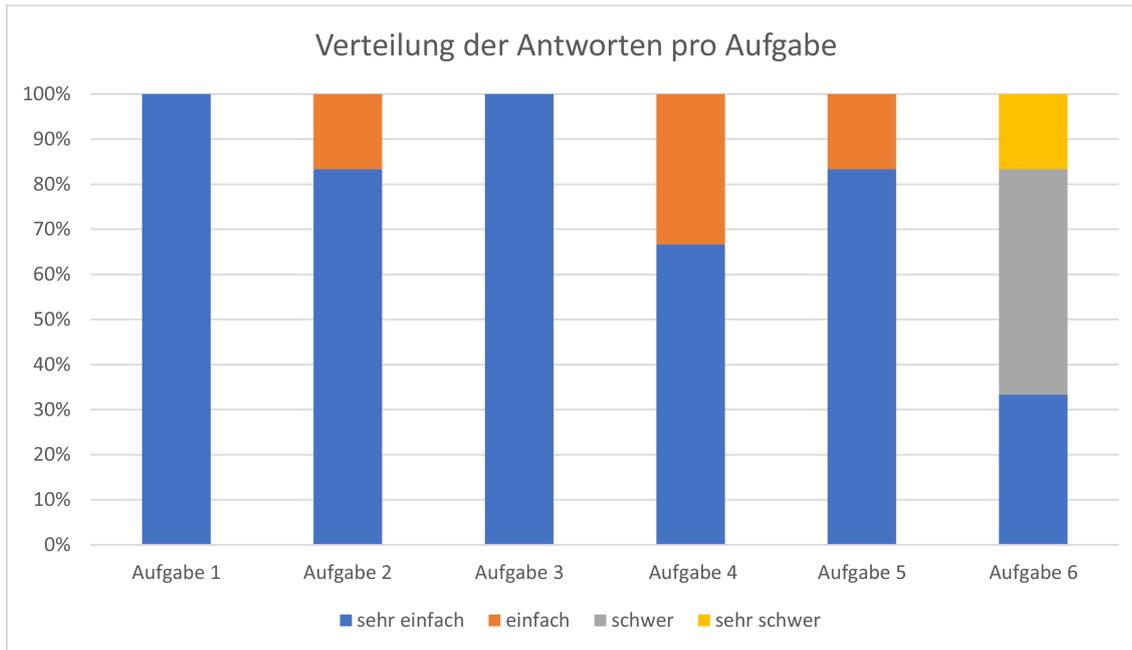


Abbildung 19: Das Diagramm zeigt den Anteil der, von den Testnutzern gewählten, Antworten bezüglich der Schwierigkeit einer Aufgabe.

Proband versuchte zuerst das Video unterhalb des Knotens einzufügen, weil die dem Text zugehörigen Pfeile nach unten gerichtet sind. Als der betreffende Nutzer das Ausbleiben eines Feedbacks seitens der Anwendung bemerkte, verbalisierte er das Fehlschlagen seines Vorhabens und fügte das Video beim zweiten Versuch erfolgreich an der richtigen Stelle ein. Wie in Abbildung 19 ersichtlich ist, schätzten alle Testteilnehmer die Aufgabe in der Anschlussbefragung als „sehr einfach“ ein.

Die Funktion, mittels *Drag-and-Drop* Dateien zu verschieben, kommt ebenfalls auf Betriebssystemebene zur Anwendung und wurde daher mit sehr hoher Wahrscheinlichkeit bereits von jedem Probanden genutzt.

Geben Sie Ihrem Projekt einen Namen.

Auch die Vergabe eines Projektnamens gelang allen Nutzern ohne Probleme. Die Testteilnehmer haben den Settings-Container auf der rechten Seite der Nutzeroberfläche als übergreifendes Element wahrgenommen und darin das passende Feld zum Benennen des Projektes zügig gefunden. Drei Probanden äußerten während der Nutzung, dass sie auf die korrekte Stelle aufgrund des beigefügten Labels aufmerksam wurden. Insgesamt schätzten die Teilnehmer diese Aufgabe als „einfach“ bis „sehr einfach“ ein.

Geben Sie an, wo das Projekt gespeichert werden soll.

Die Aufgabe, den Speicherort festzulegen, ist stark an die vorherige angelehnt und wurde daher erwartungsgemäß mit vergleichbaren Einschätzungen seitens der Versuchsteilnehmer abgeschlossen. Der sich nach Mausklick in das Textfeld öffnende Systemdialog zur Auswahl des Zielordners kam allen Teilnehmern vertraut vor. Sie nahmen diese Aufgabe als „sehr einfach“ wahr.

Sie möchten eine Frame-Extraction durchführen. Geben Sie dazu Start- und Endzeitpunkt des gewünschten Videoausschnittes an.

Eine der Grundprinzipien der Oberfläche von *VisualArchaeo3D* ist es, die für einen Arbeitsschritt benötigten Informationen und UI-Elemente dort anzuordnen, wo der Nutzer sie benötigt und erwartet. Ziel dieser Aufgabe ist es zu testen, ob dieses Konzept erfolgreich umgesetzt wurde.

Die für die *Frame-Extraction* notwendigen Eingabefelder befinden sich im dafür vorgesehenen Knoten selbst, wodurch deren Zuordnung deutlich gemacht werden soll. Die Probanden haben ohne Ausnahme die betreffenden Textfelder gefunden und ausgefüllt.

Bei der Eingabe des Start- und Endzeitpunktes für die Extraktion ist es notwendig, die gewünschte Stelle im Video zu kennen. Alle Probanden haben dazu mithilfe des in *VisualArchaeo3D* integrierten Videoplayers die passende Stelle herausgesucht. Hierbei wurde von zwei Probanden rückgemeldet, dass die Angabe im Knoten in Sekunden zu treffen ist, der Videoplayer jedoch das Zeitformat *Minuten: Sekunden* anzeigt. Drei Testteilnehmer gaben an, diese Aufgabe sei „sehr einfach“ und vier stuften sie als „einfach“ ein.

Führen Sie die Frame-Extraction aus.

Diese Aufgabe testet erneut das Verständnis des grundlegenden Aufbaus der Oberfläche. Die für die Ausführung der Pipeline zuständige Executer-Komponente wurde von fünf Nutzern sofort und von einem Nutzer nach einer geringen Pause gefunden. Die Funktion wurde dabei immer richtig zugeordnet. Besonders das aus vielen Programmen und Nutzeroberflächen bekannte Play-Symbol führte, laut Verbalisierung der Tester, zu den gewünschten Assoziationen. Auch der nebenstehende Schriftzug sowie das optische Feedback beim Überfahren der Komponente half den Nutzern. Deshalb wurde diese Aufgabe von den Nutzern als „sehr einfach“ eingestuft.

Wie können Sie einzelne Arbeitsschritte des Programms (z.B. Frame-Extraction, Keyframe-Extraction, usw.) miteinander kombinieren? Zeigen Sie dies exemplarisch.

Das Kombinieren einzelner Prozessschritte stellt die insgesamt schwierigste Aufgabe dar. Zur Lösung ist es notwendig, ein ähnliches System aus anderen Anwendungen zu kennen, oder durch das Verwenden der UI die korrekte Bedienung von selbiger „gezeigt“ zu bekommen.

Ziel der Aufgabe ist es herauszufinden, ob die Oberfläche eine dem Nutzer in der Regel unbekannt Herangehensweise zur Bewältigung einer Aufgabe durch intuitives Lernen beibringen kann.

Einen Hinweis auf die gesuchte Art und Weise der Verbindung zweier Knoten findet der Nutzer beim Blick auf die Knoten „Video Source“ und „Frame Extraction“, die von Beginn an bereits durch eine Bézierlinie miteinander verbunden sind. Sucht der Nutzer aktiv nach einer Lösung und fährt dabei mit der Maus über einen Eingangs- oder Ausgangsankerpunkt, gibt dieser durch seine Vergrößerung ein optisches Feedback und signalisiert somit die Möglichkeit der Interaktion. Findet die gewünschte Assoziation beim Nutzer nicht statt, kann er nun durch Ausprobieren sein Ziel erreichen. Letzteres trat bei zwei der Probanden auf. Nachdem ihnen zunächst nicht klar war, was zu tun ist, um zwei Knoten zu kombinieren, schlossen sie die Aufgabe nach einer kurzen Zeit des Probierens erfolgreich ab. Trotz der erfolgreichen Beendigung der Aufgabe bewerteten sie diese als „schwer“.

Ein Proband konnte die Aufgabe auch nach etwa 120 Sekunden nicht lösen. Hierbei ist anzumerken, dass dieser Proband sowohl im privaten, als auch im beruflichen Bereich nahezu keine Nutzung von Geräten mit klassischen Betriebssystemen wie *Microsoft Windows* vorweisen kann.

Zwei der Testnutzer schlossen die Aufgabe bereits nach wenigen Sekunden ab und bewerteten diese als „sehr leicht“. Beide üben einen technischen Beruf aus und nutzen sehr häufig unterschiedliche Software.

4.3.2 Auswertung der Schlussbefragung

Es folgt die Vorstellung der Ergebnisse der beiden Post-Session-Interviews, die im Anschluss an die Bearbeitung der Use-Cases durchgeführt wurden.

Visual Aesthetics of Website Inventory

Die Auswertung des *VisAWI-S* Fragebogen erfolgt durch Bildung des arithmetischen Mittelwertes aller Antworten auf der Likert-Skala. Durch die Mittlung lässt sich der errechnete Wert wieder auf die Ursprungsskala zurückführen, wodurch er, zusätzlich

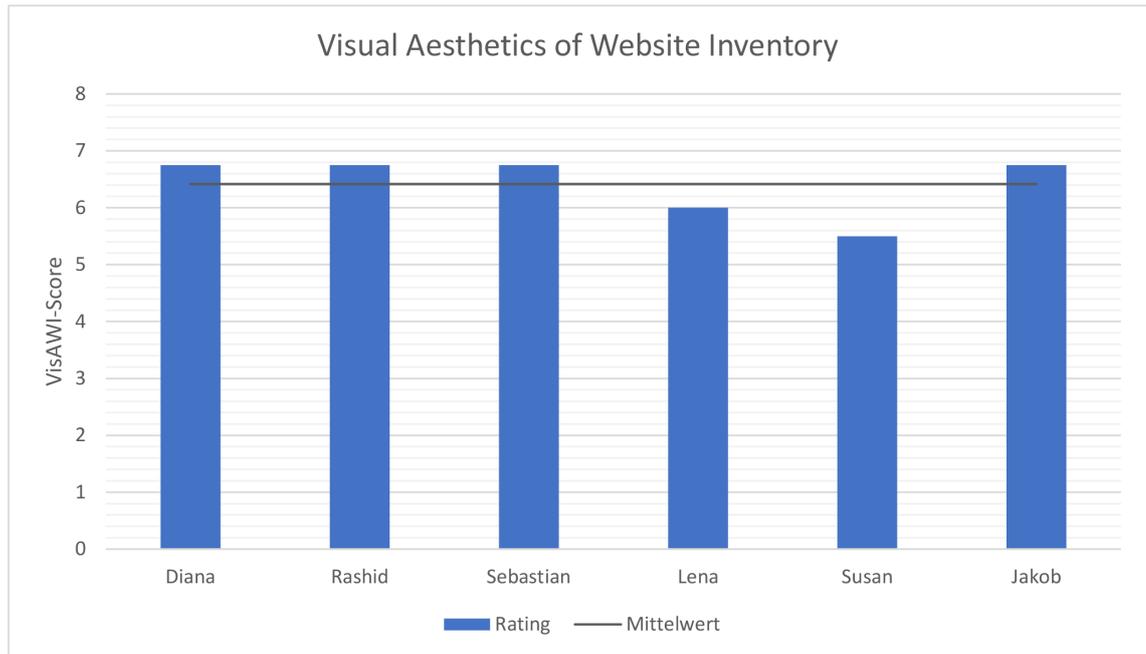


Abbildung 20: Die Auswertungsgrafik des *VisAWI-S* zeigt die jeweiligen Scores der Probanden sowie deren Mittelwert.

zum Zahlenwert, eine verbale Zuordnung erhält. Abbildung 20 führt den errechneten *VisAWI-S-Score* aller Testpersonen auf. Der mittlere *VisAWI-S-Score* ergibt sich zu 6,41, mit einem Minimalwert von 5,5 und einem Maximalwert von 6,75.

Das optische Erscheinungsbild der Benutzeroberfläche von *VisualArchaeo3D* wurde insgesamt positiv aufgenommen. Mit 6,41 liegt der *VisAWI-S-Score* nur knapp unterhalb der Höchstnote von 7. Der professionelle Eindruck, den die Software beim Nutzer hinterlassen möchte, wurde ebenfalls erreicht. Bereits bei der kleinen Stichprobengröße des vorliegenden Tests zeichnete sich ab, dass die männlichen Teilnehmer die Oberfläche tendenziell als ästhetischer wahrnehmen als die weiblichen Teilnehmerinnen.

System-Usability-Score

Zur Auswertung des *SUS-Scores* der einzelnen Probanden werden zuerst alle Antwortwerte summiert. Dabei beginnt die Zählung bei aufeinanderfolgenden Antworten jeweils von der gegenüberliegenden Seite. Danach wird der errechnete Wert mit der Zahl 2,5 multipliziert und mit der in Abbildung 21 zu sehenden Skala abgeglichen.

Bei der Befragung erreichte die getestete UI einen durchschnittlichen *SUS-Score* von 95, bei einem Minimalwert von 92,5 und einem Maximalwert von 97,5. Alle Werte liegen dabei oberhalb der *Excellent*-Grenze und sind somit als akzeptabel zu bewerten.

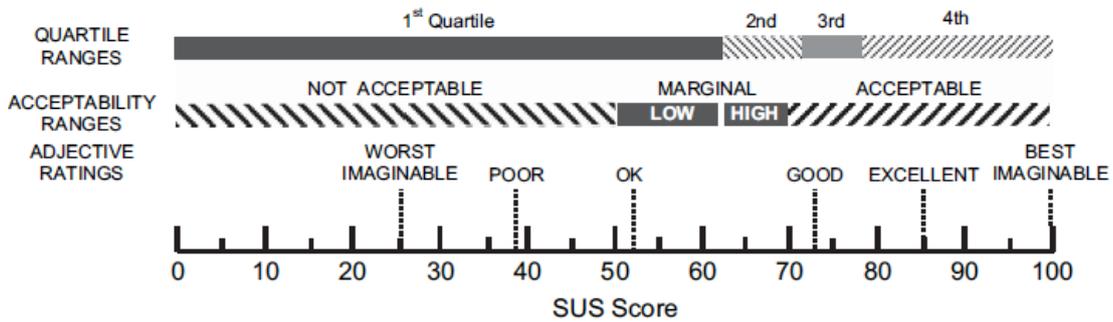


Abbildung 21: Die Abbildung zeigt, wie ein errechneter *System-Usability-Score* zu bewerten ist. Dazu ordnet die Grafik dem Zahlenwert ein Adjektiv zu und gibt Aussage über die Annehmbarkeit des Ergebnisses. Hierbei wird zwischen akzeptablen und inakzeptablen Werten unterschieden. Zwischen ihnen liegt ein Übergangsbereich. Werte zwischen 70 und 100 sind in diesem Zusammenhang als zufriedenstellend anzusehen (Abbildung aus [3]).

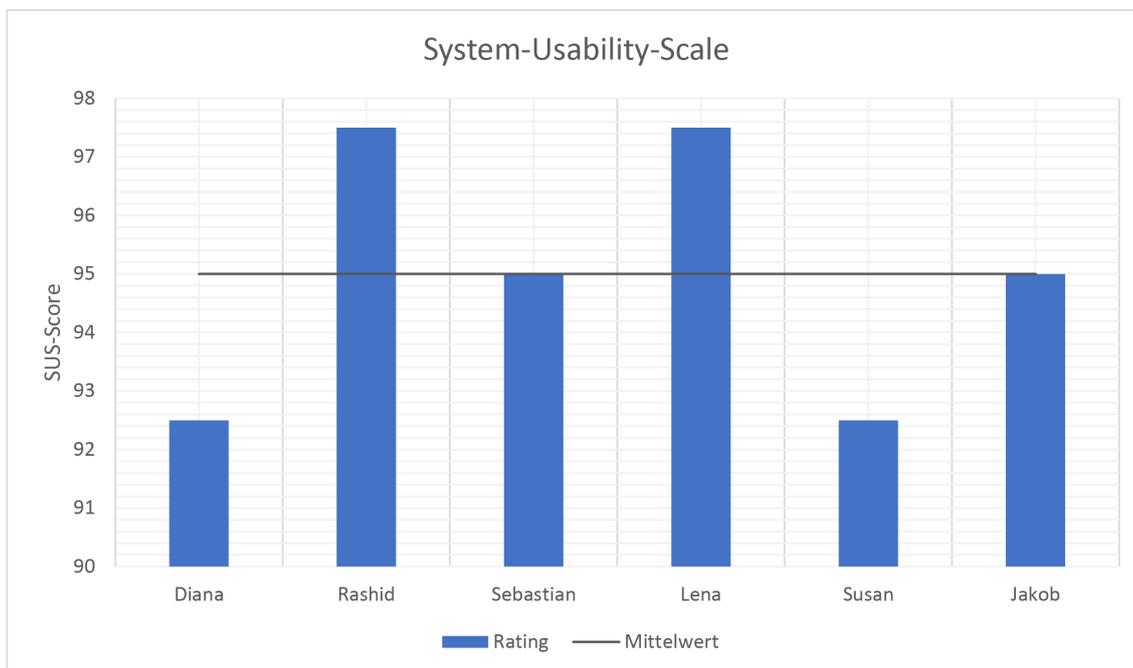


Abbildung 22: Die dargestellten *System-Usability-Scores* weisen eine hohe durchschnittliche Akzeptanz der Nutzer bezüglich *VisualArcheo3D* auf.

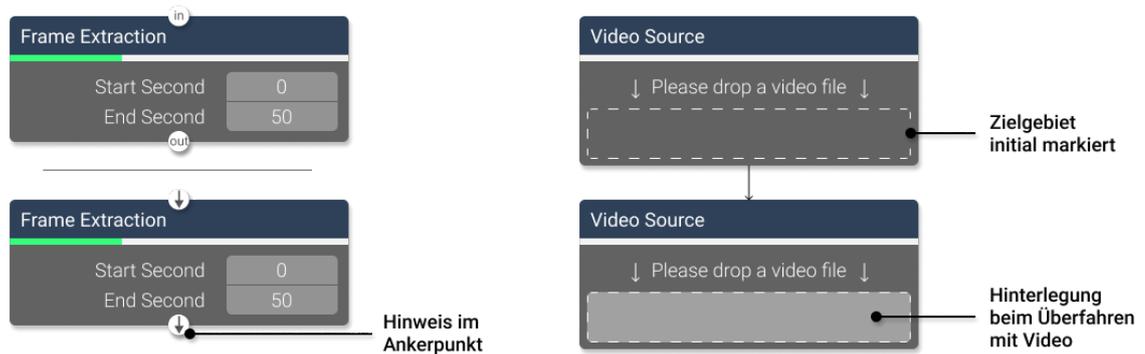


Abbildung 23: Mögliche Verbesserungen könnten beispielsweise über das Vergrößern und Beschriften der Ankerpunkte eines Knotens, oder das Markieren eines Drag-and-Drop Bereiches umgesetzt werden.

4.4 Diskussion der Ergebnisse

Das Konzept der UI, besonders im Hinblick auf das Kombinieren zweier Prozessknoten, baut maßgeblich darauf, dass der Nutzer selbstständig erkundet und aktiv ausprobiert. Dieses gewünschte Verhalten ließ sich insbesondere bei Testern beobachten, die sich zu Beginn des Tests in ihrer allgemeinen Sicherheit im Umgang mit Software als sicher oder sehr sicher einschätzten. Um Barrieren in der Nutzung von unerfahreneren Anwendern entgegenzuwirken, besteht die Möglichkeit, noch weitere Wege zu implementieren, die das Individualisieren der Prozesskette erlauben. Außerdem könnten Hinweise zum Verbinden der Knoten in den Ankerpunkten untergebracht werden. Um zu vermeiden, dass Nutzer die korrekte Stelle zum Ablegen der Videoquelle nicht finden, könnte das Zielgebiet bereits im Initialzustand markiert sein und sich beim Überfahren mit einem Video zusätzlich weiß färben. Beide Verbesserungsvarianten sind in Abbildung 23 visualisiert.

Eine Probandin hatte zuerst leichte Probleme die Metadaten im Settings-Container zum Video zuzuordnen. Zur Verdeutlichung des Zusammenhangs der beiden Elemente könnten die betroffenen Textfelder beim Einfügen eines Videos kurz aufblinken. Die Hälfte der Probanden klagte über Schwierigkeiten beim Verstehen der englischen Sprachausgabe. Da die Sprache jedoch kein Problem bei der Bedienung darstellen soll, ist es ratsam, auch andere Sprachen anzubieten. Um die Validität des Tests zu wahren, musste den Teilnehmern an dieser Stelle geholfen werden.

Insgesamt lieferte der Usability-Test zufriedenstellende Ergebnisse. Die Anwendung wurde als optisch ansprechend und insgesamt einfach bedienbar empfunden. Die festgestellten Hürden in der Bedienbarkeit der UI sind im weiteren Entwicklungsprozess einfach zu beheben.

5 Zukünftige Bedeutung graphbasierter und intuitiver Interfaces

Im Rahmen dieser Arbeit wurde untersucht, inwiefern sich ein graphbasiertes User-Interface zur Anpassung einer Videogrammetriepipeline eignet. Dabei lag der Fokus insbesondere auf der intuitiven Bedienbarkeit dieses Interfaces.

Um zu ergründen, wie eine anwenderfreundliche Nutzeroberfläche gestaltet werden kann, wurde zunächst untersucht, wie Mensch und Computer miteinander kommunizieren und welchem Modell die menschliche Informationsverarbeitung folgt. Im Kontext der Intuition sind dabei besonders unerschwinglich wahrgenommene Informationen genauer betrachtet worden.

Daraus folgte die Erstellung von *VisualArchaeo3D*, einem User-Interface für die 3D-Rekonstruktionssoftware *Archaeo3D*. Aus einem Usability-Test ging hervor, dass die in *VisualArchaeo3D* zur Anwendung kommenden Designprinzipien sowohl in der Lage sind, die gewünschten Assoziationen beim Anwender hervorzurufen, als auch das intuitive Lernen zu ermöglichen. Der Usability-Test konnte die eingangs gestellte Behauptung, dass der Mensch versucht, Mechanismen der zwischenmenschlichen Kommunikation auf die Interaktion mit Computern zu extrapolieren, bestätigen. Auch subtile Signale des User-Interfaces wurden von den Probanden wahrgenommen und führten zu erwünschten Schlussfolgerungen.

Das intuitive Erkennen von Mechanismen der UI stellte sich einerseits als effizienteste Form der Bedienung der Anwendung heraus, andererseits jedoch als die am wenigsten zuverlässigste. Grund dafür sind die ungleichen Vorkenntnisse der Anwender. Greift das Erkennen nicht, ist es wichtig, dass die Software noch weitere Wege anbietet, ihre Funktionsweise zu vermitteln. Ausprobieren stellte sich dabei als ein probates Mittel heraus, um die Nutzung eines Interfaces zu erlernen. Die UI kann den Nutzer durch subtile Hinweise, die ihm die Funktionsweise der verschiedenen Elemente verständlich machen, dazu ermutigen.

Prinzipiell stellte sich die Visualisierung der 3D-Rekonstruktionspipeline über einen Graphen als geeignet, wenn auch in ihrer Handhabung anspruchsvoll heraus. Damit sie auch bei technisch unversierten Nutzern verlässlich funktioniert, ist es ratsam, noch weitere Sicherheitsmechanismen zu implementieren, die eine erfolgreiche Bedienung garantieren.

Damit Software auch in Zukunft seine Zielgruppen zufriedenstellen und noch erweitern kann, ist es notwendig den Anwender und seine Bedürfnisse konsequent in den Fokus der Gestaltung zu rücken. Die Bedeutung graphbasierter Interfaces könnte

dabei, angesichts der bereits erfolgreich implementierten Beispiele, noch weiter steigen. Graphen, insbesondere in der Spezialform eines Flussdiagrammes, bieten schon durch ihre Benennung eine unmittelbare Verbindung zur Natur und ermöglichen dadurch nahezu jedem Menschen die potenzielle Assoziation einer Relation oder einer Abfolge. Sie sind sehr vielseitig einsetzbar und durch ihre Nähe zur Informatik gut abstrahierbar und somit leicht in Programmcode umzusetzen.

Literatur

- [1] AIGNER, M. *Diskrete Mathematik*. Vieweg Verlag, Wiesbaden, 1993.
- [2] BAERENTSEN, K. B. Intuitive interfaces. *Scandinavian Journal of Information Systems* 12, 1 (2000).
- [3] BANGOR, A., KORTUM, P. T., AND MILLER, J. T. An empirical evaluation of the system usability scale. *International Journal of Human-Computer Interaction* 24, 6 (2008), 574–594.
- [4] BLOCH, K. *Digitale Unterwasserfotografie - Equipment, Technik, Praxiswissen*. Verlagsgruppe Hüthig Jehle Rehm GmbH, Heidelberg, 2009.
- [5] BLOCK, M., DWORSKY, C., LÖW, C., FONSECA, H. S. D., GEHMLICH, B., WITTCHE, D., GÖRSCH, N., SUCHOWSKA, P., AND DUCKE, B. Underwater videogrammetry with adaptive feature detection at “See am Mondsee”, Austria. *Studies in Digital Heritage Journal* (2017).
- [6] DAHM, M. *Grundlagen der Mensch-Computer-Interaktion*. Pearson Verlag, München, 2006.
- [7] FARIN, G. *Kurven und Flächen im Computer Aided Geometric Design*. Vieweg Verlag, Braunschweig/ Wiesbaden, 1994.
- [8] FREUD, S. *Zur Dynamik der Übertragung*. Gesammelte Werke. Fischer Verlag, Frankfurt/M. Bd. VIII, S. 364-374, 1912.
- [9] GEIS, T., AND TESCH, G. *Basiswissen Usability und User Experience*. dpunkt Verlag, Heidelberg, 2019.
- [10] GLASER, C., HURTIENNE, J., AND MOHS, C. *Tagungsband UP05*. Fraunhofer Verlag, Stuttgart, 2005, ch. Intuitiv und effizient - Thesen zur Intuitivität von User Interfaces, pp. 146–153.
- [11] HEIMANN, M., AND SCHÜTZ, M. *Wie Design Wirkt - Psychologische Prinzipien erfolgreicher Gestaltung*. Rheinwerk Verlag, Bonn, 2017.
- [12] HEINECKE, A. M. *Mensch-Computer-Interaktion Basiswissen für Entwickler und Gestalter*. Springer Verlag, Heidelberg, 2004.
- [13] HOLZINGER, A. Thinking-aloud - eine Königsmethode im Usability Engineering. *OCG Journal* 31, 1 (2006), 4–5.

- [14] JACOBSEN, J., AND MAYER, L. *Praxisbuch Usability und UX*. Rheinwerk Verlag, Bonn, 2019.
- [15] JÄHNE, B. *Digitale Bildbearbeitung und Bildgewinnung*. Springer Vieweg Verlag, Berlin/ Heidelberg, 2012.
- [16] JANCOSEK, M., AND PAJDLA, T. IEEE international conference on computer vision. In *Multi-View Reconstruction Preserving Weakly-Supported Surfaces* (2011).
- [17] KOHLE, H. *Digital Humanities*. Metzler Verlag, Stuttgart, 2017, ch. Digitale Rekonstruktion und Simulation, pp. 315–327.
- [18] LORENZEN-SCHMIDT, O. Die Rolle der Usability für das Markenerlebnis. *i-com: Oldenbourg Wissenschaftsverlag GmbH, München 11*, 1 (2012), 57–62.
- [19] MEINEL, C., AND MUNDHENK, M. *Mathematische Grundlagen der Informatik*. Vieweg + Teubner Verlag, Wiesbaden, 2000.
- [20] NIELSEN, J. Enhancing the explanatory power of usability heuristics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (1994).
- [21] NIELSEN, J. Why you only need to test with 5 users. *Nielsen Norman Group* (2000).
- [22] NORMAN, D. A. *Emotional Design. Why we love (and hate) everyday things*. Basic Books Verlag, New York, 2005.
- [23] STAPELKAMP, T. *Interaction- und Interfacedesign Web-, Game-, Produkt- und Servicedesign Usability und Interface als Corporate Identity*. Springer Verlag, Heidelberg, 2010.
- [24] VINCE, J. *Mathematics for Computer Graphics*. Springer Verlag, London, 2001.

Anhang

Usability-Test

VisualArchaeo3D

Alter, Beruf: 30, Archiventw.

Geschlecht: männlich

Testperson (Kürzel): Sebastian

Datum, Uhrzeit: 27.03 12⁰⁰

Einführung (Jens Jacobsen | www.benutzerfreun.de)

„Diese Sitzung **dauert etwa 30 Minuten**. Dabei werde ich Sie bitten, ein paar Aufgaben zu bearbeiten. Danach stelle ich Ihnen einige Fragen, um Ihre Meinung zu erfahren, und um zu verstehen, wie Sie vorgegangen sind.“

„Wir testen die Anwendung und wie gut diese ist. **Sie können gar nichts falsch machen** – wenn Sie auf ein Problem stoßen, ist das für uns sehr interessant, und wir können die Anwendung dann verbessern.“

„Sie können jederzeit **unbesorgt Ihre Meinung äußern**. Meine Aufgabe ist es, Feedback aufzunehmen und weiterzugeben. Positive und negative Anmerkungen von Ihnen sind für uns gleich interessant.“

„Ganz wichtig: Uns interessiert sehr, **was Sie denken**, während Sie mit der [Software] arbeiten. Nur so kann ich verstehen, warum Sie tun, was Sie tun. Bevor Sie also klicken, sagen Sie mir bitte jeweils zuerst, was Sie vorhaben. Und sagen Sie mir, was Sie erwarten, dass passiert.“

„Zum Beispiel so: Wenn ich das Menü ansehe, dann würde ich auf „Startseite“ klicken. Ich würde erwarten, dass ich dann zur Startseite komme.“

„Ich werde Sie nun bitten, ein paar Dinge mit der Anwendung zu tun. Machen Sie das genau so, wie Sie das auch tun würden, wenn ich nicht dabei wäre.“

Sie können mich jederzeit nach etwas fragen. Manchmal werde ich aber nicht antworten, weil mich interessiert, ob die [Software] es ermöglicht, dass man manche Dinge ohne Hilfe dort tun kann.

Haben Sie noch Fragen? Dann stellen Sie sie mir jetzt gern.“

Ausgangssituation:

Sie haben ein Video erstellt und möchten aus diesem ein 3D-Modell generieren. Bei der Recherche im Internet sind sie auf das Programm VisualArchaeo3D gestoßen, das damit wirbt, Keyframes aus einem Video zu extrahieren.

Analyse des Probanden/ Selbsteinschätzung

	sehr sicher	sicher	weniger sicher	unsicher
Wie sicher schätzen Sie sich selbst im Umgang mit Software ein?	X			

Sind Sie bereits mit 3D-Rekonstruktion in Berührung gekommen? ja nein

Welche technischen Geräte nutzen Sie beruflich und privat?

Gerät	beruflich	privat
Desktop-PC		
Laptop	X	X
Smartphone	X	X
Tablet		
Andere:		

Welche Software nutzen sie regelmäßig beruflich und privat?

beruflich	privat
SPS, Python, MATLAB	

Aufgaben (Use Cases)

Wie einfach/schwer fanden Sie folgende Aufgaben?	sehr einfach 1	2	3	sehr schwer 4
Fügen Sie das gewünschte Video in VisualArchaeo3D ein.	X			
Geben Sie Ihrem Projekt einen Namen.	X			
Geben Sie an, wo das Projekt gespeichert werden soll.	X			
Sie möchten eine Frame-Extraction durchführen. Geben Sie dazu Start und Endzeitpunkt des gewünschten Videoausschnittes an.	X			
Führen Sie die Frame-Extraction aus.	X			
Wie können einzelne Arbeitsschritte des Programmes (z.B. Frame Extraction, Keyframe Extraction usw.) miteinander kombiniert werden? Zeigen Sie dies exemplarisch.	X			

folgt der Beschriftung
versucht Pfad
zu kopieren
Dialog öffnet
sich

Visual Aesthetics of Websites Inventory, nach Moshagen und Thielsch, 2013

	stimme gar nicht zu	stimme nicht zu	stimme eher nicht zu	neutral	stimme eher zu	stimme zu	stimme voll zu
Auf der Seite passt alles zusammen.							X
Das Layout ist angenehm vielseitig.						X	
Die farbliche Gesamtgestaltung wirkt attraktiv.							X
Das Layout ist professionell.							X

VisAwi-Score: $27 : 4 = \underline{\underline{6,75}}$

System Usability Scale nach Brooke, 1996

	lehne völlig ab	0	1	2	3	4	stimme völlig zu
Ich denke, ich würde die Software regelmäßig nutzen.					X		
Die Software erscheint mir unnötig kompliziert.	X						
Ich finde die Software ist einfach zu benutzen.							X
Ich denke, ich bräuchte technische Unterstützung um die Software nutzen zu können.	X						
Ich finde, dass die verschiedenen Funktionen der Software gut integriert sind.							X
Die Software erscheint mir zu uneinheitlich.	X						
Ich glaube, dass die meisten Leute die Benutzung der Software schnell erlernen können.					X		
Die Software erscheint mir sehr umständlich zu benutzen.	X						
Ich fühle mich bei der Benutzung der Software sehr sicher.							X
Ich musste einiges lernen, um mit der Software zurecht zu kommen.	X						

Sus-Score: $38 : 25 = \underline{\underline{95}}$

Feedback des Probanden

Haben Sie noch allgemeine Anmerkungen zu diesem Test?

Wie würden Sie das Ihnen vorliegende Programm verbessern?

- Feedback im Prozedur "allumfänglich"

Vielen Dank für Ihre Teilnahme.

Usability-Test

VisualArchaeo3D

Alter, Beruf:

36, Erziehungswissenschaftlerin

Geschlecht:

weiblich

Testperson (Kürzel):

Diana

Datum, Uhrzeit:

27.03. 10³⁵

Einführung (Jens Jacobsen | www.benutzerfreun.de)

„Diese Sitzung **dauert etwa 30 Minuten**. Dabei werde ich Sie bitten, ein paar Aufgaben zu bearbeiten. Danach stelle ich Ihnen einige Fragen, um Ihre Meinung zu erfahren, und um zu verstehen, wie Sie vorgegangen sind.“

„Wir testen die Anwendung und wie gut diese ist. **Sie können gar nichts falsch machen** – wenn Sie auf ein Problem stoßen, ist das für uns sehr interessant, und wir können die Anwendung dann verbessern.“

„Sie können jederzeit **unbesorgt Ihre Meinung äußern**. Meine Aufgabe ist es, Feedback aufzunehmen und weiterzugeben. Positive und negative Anmerkungen von Ihnen sind für uns gleich interessant.“

„Ganz wichtig: Uns interessiert sehr, **was Sie denken**, während Sie mit der [Software] arbeiten. Nur so kann ich verstehen, warum Sie tun, was Sie tun. Bevor Sie also klicken, sagen Sie mir bitte jeweils zuerst, was Sie vorhaben. Und sagen Sie mir, was Sie erwarten, dass passiert.“

„Zum Beispiel so: Wenn ich das Menü ansehe, dann würde ich auf „Startseite“ klicken. Ich würde erwarten, dass ich dann zur Startseite komme.“

„Ich werde Sie nun bitten, ein paar Dinge mit der Anwendung zu tun. Machen Sie das genau so, wie Sie das auch tun würden, wenn ich nicht dabei wäre.“

Sie können mich jederzeit nach etwas fragen. Manchmal werde ich aber nicht antworten, weil mich interessiert, ob die [Software] es ermöglicht, dass man manche Dinge ohne Hilfe dort tun kann.

Haben Sie noch Fragen? Dann stellen Sie sie mir jetzt gern.“

Ausgangssituation:

Sie haben ein Video erstellt und möchten aus diesem ein 3D-Modell generieren. Bei der Recherche im Internet sind sie auf das Programm VisualArchaeo3D gestoßen, das damit wirbt, Keyframes aus einem Video zu extrahieren.

Analyse des Probanden/ Selbsteinschätzung

	sehr sicher	sicher	weniger sicher	unsicher
Wie sicher schätzen Sie sich selbst im Umgang mit Software ein?		X		

Sind Sie bereits mit 3D-Rekonstruktion in Berührung gekommen? ja nein

Welche technischen Geräte nutzen Sie beruflich und privat?

Gerät	beruflich	privat
Desktop-PC	X	
Laptop	X	X
Smartphone		X
Tablet		X
Andere: Drucker		

Welche Software nutzen sie regelmäßig beruflich und privat?

beruflich	privat
Typo3, SharePoint MS. Office	

Aufgaben (Use Cases)

Wie einfach/schwer fanden Sie folgende Aufgaben?	sehr einfach 1	2	3	sehr schwer 4
Fügen Sie das gewünschte Video in VisualArchaeo3D ein.	X			
Geben Sie Ihrem Projekt einen Namen.	X			
Geben Sie an, wo das Projekt gespeichert werden soll.	X			
Sie möchten eine Frame-Extraction durchführen. Geben Sie dazu Start und Endzeitpunkt des gewünschten Videoausschnittes an.		X		
Führen Sie die Frame-Extraction aus.		X		
Wie können einzelne Arbeitsschritte des Programmes (z.B. Frame Extraction, Keyframe Extraction usw.) miteinander kombiniert werden? Zeigen Sie dies exemplarisch.			X	

"drag-and-drop"
"da wo es steht"
Feld mit pseudonymen
Namen ausgewählt
"click to open"
wie oben
Sekunden-
format

↳ Kontext unbekannt

↓
durch Ausprobieren
gefunden

Visual Aesthetics of Websites Inventory, nach Moshagen und Thielsch, 2013

	stimme gar nicht zu	stimme nicht zu	stimme eher nicht zu	neutral	stimme eher zu	stimme zu	stimme voll zu
Auf der Seite passt alles zusammen.							X
Das Layout ist angenehm vielseitig.							X
Die farbliche Gesamtgestaltung wirkt attraktiv.						X	
Das Layout ist professionell.							X

VisAWI-Score: $27 \cdot 4 = \underline{108}$

System Usability Scale nach Brooke, 1996

	lehne völlig ab	0	1	2	3	4	stimme völlig zu
Ich denke, ich würde die Software regelmäßig nutzen.							X
Die Software erscheint mir unnötig kompliziert.			X				
Ich finde die Software ist einfach zu benutzen.							X
Ich denke, ich bräuchte technische Unterstützung um die Software nutzen zu können.	X						
Ich finde, dass die verschiedenen Funktionen der Software gut integriert sind.							X
Die Software erscheint mir zu uneinheitlich.	X						
Ich glaube, dass die meisten Leute die Benutzung der Software schnell erlernen können.					X		
Die Software erscheint mir sehr umständlich zu benutzen.	X						
Ich fühle mich bei der Benutzung der Software sehr sicher.					X		
Ich musste einiges lernen, um mit der Software zurecht zu kommen.	X						

SUS-Score: $37 \cdot 25 = \underline{925}$

Feedback des Probanden

Haben Sie noch allgemeine Anmerkungen zu diesem Test?

Wie würden Sie das Ihnen vorliegende Programm verbessern?

• Hintergrund heller?

Vielen Dank für Ihre Teilnahme.

Usability-Test

VisualArchaeo3D

Alter, Beruf: 30, Polizist

Geschlecht: männlich

Testperson (Kürzel): Jakob

Datum, Uhrzeit: 27.03., 14:00

Einführung (Jens Jacobsen | www.benutzerfreun.de)

„Diese Sitzung **dauert etwa 30 Minuten**. Dabei werde ich Sie bitten, ein paar Aufgaben zu bearbeiten. Danach stelle ich Ihnen einige Fragen, um Ihre Meinung zu erfahren, und um zu verstehen, wie Sie vorgegangen sind.“

„Wir testen die Anwendung und wie gut diese ist. **Sie können gar nichts falsch machen** – wenn Sie auf ein Problem stoßen, ist das für uns sehr interessant, und wir können die Anwendung dann verbessern.“

„Sie können jederzeit **unbesorgt Ihre Meinung äußern**. Meine Aufgabe ist es, Feedback aufzunehmen und weiterzugeben. Positive und negative Anmerkungen von Ihnen sind für uns gleich interessant.“

„Ganz wichtig: Uns interessiert sehr, **was Sie denken**, während Sie mit der [Software] arbeiten. Nur so kann ich verstehen, warum Sie tun, was Sie tun. Bevor Sie also klicken, sagen Sie mir bitte jeweils zuerst, was Sie vorhaben. Und sagen Sie mir, was Sie erwarten, dass passiert.“

„Zum Beispiel so: Wenn ich das Menü ansehe, dann würde ich auf „Startseite“ klicken. Ich würde erwarten, dass ich dann zur Startseite komme.“

„Ich werde Sie nun bitten, ein paar Dinge mit der Anwendung zu tun. Machen Sie das genau so, wie Sie das auch tun würden, wenn ich nicht dabei wäre.“

Sie können mich jederzeit nach etwas fragen. Manchmal werde ich aber nicht antworten, weil mich interessiert, ob die [Software] es ermöglicht, dass man manche Dinge ohne Hilfe dort tun kann.

Haben Sie noch Fragen? Dann stellen Sie sie mir jetzt gern.“

Ausgangssituation:

Sie haben ein Video erstellt und möchten aus diesem ein 3D-Modell generieren. Bei der Recherche im Internet sind sie auf das Programm VisualArchaeo3D gestoßen, das damit wirbt, Keyframes aus einem Video zu extrahieren.

Analyse des Probanden/ Selbsteinschätzung

	sehr sicher	sicher	weniger sicher	unsicher
Wie sicher schätzen Sie sich selbst im Umgang mit Software ein?		X		

Sind Sie bereits mit 3D-Rekonstruktion in Berührung gekommen? ja nein

Welche technischen Geräte nutzen Sie beruflich und privat?

Gerät	beruflich	privat
Desktop-PC	X	
Laptop		X
Smartphone		X
Tablet		X
Andere: Funkgerät	X	

Welche Software nutzen sie regelmäßig beruflich und privat?

beruflich	privat
MS Office	YouTube
	Twitter

Aufgaben (Use Cases)

Wie einfach/schwer fanden Sie folgende Aufgaben?	sehr einfach 1	2	3	sehr schwer 4
Fügen Sie das gewünschte Video in VisualArcheo3D ein.	X			
Geben Sie Ihrem Projekt einen Namen.	X			
Geben Sie an, wo das Projekt gespeichert werden soll.	X			
Sie möchten eine Frame-Extraction durchführen. Geben Sie dazu Start und Endzeitpunkt des gewünschten Videoausschnittes an.	X			
Führen Sie die Frame-Extraction aus.	X			
Wie können einzelne Arbeitsschritte des Programmes (z.B. Frame Extraction, Keyframe Extraction usw.) miteinander kombiniert werden? Zeigen Sie dies exemplarisch.				X

wollte unterhalb des Kofers auftragen

Videolänge sofort erfahren

Visual Aesthetics of Websites Inventory, nach Moshagen und Thielsch, 2013

	stimme gar nicht zu	stimme nicht zu	stimme eher nicht zu	neutral	stimme eher zu	stimme zu	stimme voll zu
Auf der Seite passt alles zusammen.							X
Das Layout ist angenehm vielseitig.						X	
Die farbliche Gesamtgestaltung wirkt attraktiv.							X
Das Layout ist professionell.							X

VisAWI - Score $27:4 = \underline{\underline{6,75}}$

System Usability Scale nach Brooke, 1996

	lehne völlig ab 0	1	2	3	stimme völlig zu 4
Ich denke, ich würde die Software regelmäßig nutzen.					X
Die Software erscheint mir unnötig kompliziert.	X				
Ich finde die Software ist einfach zu benutzen.				X	
Ich denke, ich bräuchte technische Unterstützung um die Software nutzen zu können.	X				
Ich finde, dass die verschiedenen Funktionen der Software gut integriert sind.					X
Die Software erscheint mir zu uneinheitlich.	X				
Ich glaube, dass die meisten Leute die Benutzung der Software schnell erlernen können.					X
Die Software erscheint mir sehr umständlich zu benutzen.	X				
Ich fühle mich bei der Benutzung der Software sehr sicher.				X	
Ich musste einiges lernen, um mit der Software zurecht zu kommen.	X				

SUS-Score: $36:25 = \underline{\underline{95}}$

Feedback des Probanden

Haben Sie noch allgemeine Anmerkungen zu diesem Test?

Wie würden Sie das Ihnen vorliegende Programm verbessern?

- "Abwurfzeit" größer machen

Vielen Dank für Ihre Teilnahme.

Usability-Test

VisualArchaeo3D

Alter, Beruf: 28, Designerin

Geschlecht: weiblich

Testperson (Kürzel): Lena

Datum, Uhrzeit: 27.03, 12:20

Einführung (Jens Jacobsen | www.benutzerfreun.de)

„Diese Sitzung **dauert etwa 30 Minuten**. Dabei werde ich Sie bitten, ein paar Aufgaben zu bearbeiten. Danach stelle ich Ihnen einige Fragen, um Ihre Meinung zu erfahren, und um zu verstehen, wie Sie vorgegangen sind.“

„Wir testen die Anwendung und wie gut diese ist. **Sie können gar nichts falsch machen** – wenn Sie auf ein Problem stoßen, ist das für uns sehr interessant, und wir können die Anwendung dann verbessern.“

„Sie können jederzeit **unbesorgt Ihre Meinung äußern**. Meine Aufgabe ist es, Feedback aufzunehmen und weiterzugeben. Positive und negative Anmerkungen von Ihnen sind für uns gleich interessant.“

„Ganz wichtig: Uns interessiert sehr, **was Sie denken**, während Sie mit der [Software] arbeiten. Nur so kann ich verstehen, warum Sie tun, was Sie tun. Bevor Sie also klicken, sagen Sie mir bitte jeweils zuerst, was Sie vorhaben. Und sagen Sie mir, was Sie erwarten, dass passiert.“

„Zum Beispiel so: Wenn ich das Menü ansehe, dann würde ich auf „Startseite“ klicken. Ich würde erwarten, dass ich dann zur Startseite komme.“

„Ich werde Sie nun bitten, ein paar Dinge mit der Anwendung zu tun. Machen Sie das genau so, wie Sie das auch tun würden, wenn ich nicht dabei wäre.“

Sie können mich jederzeit nach etwas fragen. Manchmal werde ich aber nicht antworten, weil mich interessiert, ob die [Software] es ermöglicht, dass man manche Dinge ohne Hilfe dort tun kann.

Haben Sie noch Fragen? Dann stellen Sie sie mir jetzt gern.“

Ausgangssituation:

Sie haben ein Video erstellt und möchten aus diesem ein 3D-Modell generieren. Bei der Recherche im Internet sind sie auf das Programm VisualArchaeo3D gestoßen, das damit wirbt, Keyframes aus einem Video zu extrahieren.

Analyse des Probanden/ Selbsteinschätzung

	sehr sicher	sicher	weniger sicher	unsicher
Wie sicher schätzen Sie sich selbst im Umgang mit Software ein?			X	

Sind Sie bereits mit 3D-Rekonstruktion in Berührung gekommen? ja nein

Welche technischen Geräte nutzen Sie beruflich und privat?

Gerät	beruflich	privat
Desktop-PC	X	X
Laptop	X	X
Smartphone	X	X
Tablet		
Andere: Touch-Kassensystem		

Welche Software nutzen sie regelmäßig beruflich und privat?

beruflich	privat
SAP	MS-office

Aufgaben (Use Cases)

Wie einfach/schwer fanden Sie folgende Aufgaben?	sehr einfach			sehr schwer
	1	2	3	4
Fügen Sie das gewünschte Video in VisualArcheo3D ein.	X			
Geben Sie Ihrem Projekt einen Namen.		X		
Geben Sie an, wo das Projekt gespeichert werden soll.	X			
Sie möchten eine Frame-Extraction durchführen. Geben Sie dazu Start und Endzeitpunkt des gewünschten Videoausschnittes an.	X			
Führen Sie die Frame-Extraction aus.	X			
Wie können einzelne Arbeitsschritte des Programmes (z.B. Frame Extraction, Keyframe Extraction usw.) miteinander kombiniert werden? Zeigen Sie dies exemplarisch.			X	

Ento gedrückt

Visual Aesthetics of Websites Inventory, nach Moshagen und Thielsch, 2013

	stimme gar nicht zu	stimme nicht zu	stimme eher nicht zu	neutral	stimme eher zu	stimme zu	stimme voll zu
Auf der Seite passt alles zusammen.							X
Das Layout ist angenehm vielseitig.						X	
Die farbliche Gesamtgestaltung wirkt attraktiv.				X			
Das Layout ist professionell.							X

Visual-Score: $24:4 = 6$

System Usability Scale nach Brooke, 1996

	lehne völlig ab	0	1	2	3	4	stimme völlig zu
Ich denke, ich würde die Software regelmäßig nutzen.							X
Die Software erscheint mir unnötig kompliziert.	X						
Ich finde die Software ist einfach zu benutzen.							X
Ich denke, ich bräuchte technische Unterstützung um die Software nutzen zu können.	X						
Ich finde, dass die verschiedenen Funktionen der Software gut integriert sind.							X
Die Software erscheint mir zu uneinheitlich.	X						
Ich glaube, dass die meisten Leute die Benutzung der Software schnell erlernen können.							X
Die Software erscheint mir sehr umständlich zu benutzen.	X						
Ich fühle mich bei der Benutzung der Software sehr sicher.						X	
Ich musste einiges lernen, um mit der Software zurecht zu kommen.	X						

SUS-Score: $39:25 = 97,5$

Feedback des Probanden

Haben Sie noch allgemeine Anmerkungen zu diesem Test?

Wie würden Sie das Ihnen vorliegende Programm verbessern?

- Programm auf deutsch

Vielen Dank für Ihre Teilnahme.

Usability-Test

VisualArchaeo3D

Alter, Beruf: 37, Banker

Geschlecht: männlich

Testperson (Kürzel): Rashid

Datum, Uhrzeit: 27.03. 11⁰⁷

Einführung (Jens Jacobsen | www.benutzerfreun.de)

„Diese Sitzung **dauert etwa 30 Minuten**. Dabei werde ich Sie bitten, ein paar Aufgaben zu bearbeiten. Danach stelle ich Ihnen einige Fragen, um Ihre Meinung zu erfahren, und um zu verstehen, wie Sie vorgegangen sind.“

„Wir testen die Anwendung und wie gut diese ist. **Sie können gar nichts falsch machen** – wenn Sie auf ein Problem stoßen, ist das für uns sehr interessant, und wir können die Anwendung dann verbessern.“

„Sie können jederzeit **unbesorgt Ihre Meinung äußern**. Meine Aufgabe ist es, Feedback aufzunehmen und weiterzugeben. Positive und negative Anmerkungen von Ihnen sind für uns gleich interessant.“

„Ganz wichtig: Uns interessiert sehr, **was Sie denken**, während Sie mit der [Software] arbeiten. Nur so kann ich verstehen, warum Sie tun, was Sie tun. Bevor Sie also klicken, sagen Sie mir bitte jeweils zuerst, was Sie vorhaben. Und sagen Sie mir, was Sie erwarten, dass passiert.“

„Zum Beispiel so: Wenn ich das Menü ansehe, dann würde ich auf „Startseite“ klicken. Ich würde erwarten, dass ich dann zur Startseite komme.“

„Ich werde Sie nun bitten, ein paar Dinge mit der Anwendung zu tun. Machen Sie das genau so, wie Sie das auch tun würden, wenn ich nicht dabei wäre.“

Sie können mich jederzeit nach etwas fragen. Manchmal werde ich aber nicht antworten, weil mich interessiert, ob die [Software] es ermöglicht, dass man manche Dinge ohne Hilfe dort tun kann.

Haben Sie noch Fragen? Dann stellen Sie sie mir jetzt gern.“

Ausgangssituation:

Sie haben ein Video erstellt und möchten aus diesem ein 3D-Modell generieren. Bei der Recherche im Internet sind sie auf das Programm VisualArchaeo3D gestoßen, das damit wirbt, Keyframes aus einem Video zu extrahieren.

Analyse des Probanden/ Selbsteinschätzung

	sehr sicher	sicher	weniger sicher	unsicher
Wie sicher schätzen Sie sich selbst im Umgang mit Software ein?	X			

Sind Sie bereits mit 3D-Rekonstruktion in Berührung gekommen? ja nein

Welche technischen Geräte nutzen Sie beruflich und privat?

Gerät	beruflich	privat
Desktop-PC	X	
Laptop	X	
Smartphone	X	
Tablet	X	
Andere:		

Welche Software nutzen sie regelmäßig beruflich und privat?

beruflich	privat
<ul style="list-style-type: none"> • Web-Design • 3D-Studio Max 	Video-Editing

Aufgaben (Use Cases)

Wie einfach/schwer fanden Sie folgende Aufgaben?	sehr einfach 1	2	3	sehr schwer 4
Fügen Sie das gewünschte Video in VisualArcheo3D ein.	X			
Geben Sie Ihrem Projekt einen Namen.	X			
Geben Sie an, wo das Projekt gespeichert werden soll.	X			
Sie möchten eine Frame-Extraction durchführen. Geben Sie dazu Start und Endzeitpunkt des gewünschten Videoausschnittes an.	X			
Führen Sie die Frame-Extraction aus.	X			
Wie können einzelne Arbeitsschritte des Programmes (z.B. Frame Extraction, Keyframe Extraction usw.) miteinander kombiniert werden? Zeigen Sie dies exemplarisch.	X			

Bemerkungen
 Drag-and-Drop
 • Cursor bei Enter aus Inputfeld raus
 • nicht Rückmeld. positiv hervor

Visual Aesthetics of Websites Inventory, nach Moshagen und Thielsch, 2013

	stimme gar nicht zu	stimme nicht zu	stimme eher nicht zu	neutral	stimme eher zu	stimme zu	stimme voll zu
Auf der Seite passt alles zusammen.							X
Das Layout ist angenehm vielseitig.							X
Die farbliche Gesamtgestaltung wirkt attraktiv.						X	
Das Layout ist professionell.							X

VisAWI-Score: $27:4 = \underline{\underline{6,75}}$

System Usability Scale nach Brooke, 1996

	lehne völlig ab 0	1	2	3	stimme völlig zu 4
Ich denke, ich würde die Software regelmäßig nutzen.				X	
Die Software erscheint mir unnötig kompliziert.	X				
Ich finde die Software ist einfach zu benutzen.					X
Ich denke, ich bräuchte technische Unterstützung um die Software nutzen zu können.	X				
Ich finde, dass die verschiedenen Funktionen der Software gut integriert sind.					X
Die Software erscheint mir zu uneinheitlich.	X				
Ich glaube, dass die meisten Leute die Benutzung der Software schnell erlernen können.					X
Die Software erscheint mir sehr umständlich zu benutzen.	X				
Ich fühle mich bei der Benutzung der Software sehr sicher.					X
Ich musste einiges lernen, um mit der Software zurecht zu kommen.	X				

SUS-Score: $43:25 = \underline{\underline{97,5}}$

Feedback des Probanden

Haben Sie noch allgemeine Anmerkungen zu diesem Test?

Wie würden Sie das Ihnen vorliegende Programm verbessern?

- empfiehlt Grid im Hintergrund als "Edit-Mode"
- wünscht sich Tastenkombinationen

Vielen Dank für Ihre Teilnahme.

Usability-Test

VisualArchaeo3D

Alter, Beruf: 52, Pädagogin

Geschlecht: weiblich

Testperson (Kürzel): Susan

Datum, Uhrzeit: 27.03., 12⁴⁷

Einführung (Jens Jacobsen | www.benutzerfreun.de)

„Diese Sitzung **dauert etwa 30 Minuten**. Dabei werde ich Sie bitten, ein paar Aufgaben zu bearbeiten. Danach stelle ich Ihnen einige Fragen, um Ihre Meinung zu erfahren, und um zu verstehen, wie Sie vorgegangen sind.“

„Wir testen die Anwendung und wie gut diese ist. **Sie können gar nichts falsch machen** – wenn Sie auf ein Problem stoßen, ist das für uns sehr interessant, und wir können die Anwendung dann verbessern.“

„Sie können jederzeit **unbesorgt Ihre Meinung äußern**. Meine Aufgabe ist es, Feedback aufzunehmen und weiterzugeben. Positive und negative Anmerkungen von Ihnen sind für uns gleich interessant.“

„Ganz wichtig: Uns interessiert sehr, **was Sie denken**, während Sie mit der [Software] arbeiten. Nur so kann ich verstehen, warum Sie tun, was Sie tun. Bevor Sie also klicken, sagen Sie mir bitte jeweils zuerst, was Sie vorhaben. Und sagen Sie mir, was Sie erwarten, dass passiert.“

„Zum Beispiel so: Wenn ich das Menü ansehe, dann würde ich auf „Startseite“ klicken. Ich würde erwarten, dass ich dann zur Startseite komme.“

„Ich werde Sie nun bitten, ein paar Dinge mit der Anwendung zu tun. Machen Sie das genau so, wie Sie das auch tun würden, wenn ich nicht dabei wäre.“

Sie können mich jederzeit nach etwas fragen. Manchmal werde ich aber nicht antworten, weil mich interessiert, ob die [Software] es ermöglicht, dass man manche Dinge ohne Hilfe dort tun kann.

Haben Sie noch Fragen? Dann stellen Sie sie mir jetzt gern.“

Ausgangssituation:

Sie haben ein Video erstellt und möchten aus diesem ein 3D-Modell generieren. Bei der Recherche im Internet sind sie auf das Programm VisualArchaeo3D gestoßen, das damit wirbt, Keyframes aus einem Video zu extrahieren.

Analyse des Probanden/ Selbsteinschätzung

	sehr sicher	sicher	weniger sicher	unsicher
Wie sicher schätzen Sie sich selbst im Umgang mit Software ein?			X	

Sind Sie bereits mit 3D-Rekonstruktion in Berührung gekommen? ja nein

Welche technischen Geräte nutzen Sie beruflich und privat?

Gerät	beruflich	privat
Desktop-PC	X	X
Laptop	X	X
Smartphone	X	X
Tablet		
Andere:		

Welche Software nutzen sie regelmäßig beruflich und privat?

beruflich	privat
Excel, Word, Powerpoint	Youtube
	Online-Banking
Personal-Verwaltungssoftware	

Aufgaben (Use Cases)

Wie einfach/schwer fanden Sie folgende Aufgaben?	sehr einfach			sehr schwer
	1	2	3	4
Fügen Sie das gewünschte Video in VisualArcheo3D ein.	X			
Geben Sie Ihrem Projekt einen Namen.	X			
Geben Sie an, wo das Projekt gespeichert werden soll.	X			
Sie möchten eine Frame-Extraction durchführen. Geben Sie dazu Start und Endzeitpunkt des gewünschten Videoausschnittes an.		X		
Führen Sie die Frame-Extraction aus.	X			
Wie können einzelne Arbeitsschritte des Programmes (z.B. Frame Extraction, Keyframe Extraction usw.) miteinander kombiniert werden? Zeigen Sie dies exemplarisch.			X	

Orientierung an Beschriftung wie aus anderen Beispielen bekannt Zuordnung der Metadaten ↳ Sprache ↳ techn. Verständnis

↓
durch Probieren gefunden

Visual Aesthetics of Websites Inventory, nach Moshagen und Thielsch, 2013

	stimme gar nicht zu	stimme nicht zu	stimme eher nicht zu	neutral	stimme eher zu	stimme zu	stimme voll zu
Auf der Seite passt alles zusammen.						X	
Das Layout ist angenehm vielseitig.		X					
Die farbliche Gesamtgestaltung wirkt attraktiv.							X
Das Layout ist professionell.							X

vis AWI - Score : $22 : 4 = \underline{5,5}$

System Usability Scale nach Brooke, 1996

	lehne völlig ab				stimme völlig zu
	0	1	2	3	4
Ich denke, ich würde die Software regelmäßig nutzen.					X ₄
Die Software erscheint mir unnötig kompliziert.	X ₀				
Ich finde die Software ist einfach zu benutzen.					X ₄
Ich denke, ich bräuchte technische Unterstützung um die Software nutzen zu können.		X ₁			
Ich finde, dass die verschiedenen Funktionen der Software gut integriert sind.					X ₄
Die Software erscheint mir zu uneinheitlich.	X ₀				
Ich glaube, dass die meisten Leute die Benutzung der Software schnell erlernen können.					X ₄
Die Software erscheint mir sehr umständlich zu benutzen.	X ₀				
Ich fühle mich bei der Benutzung der Software sehr sicher.				X ₃	
Ich musste einiges lernen, um mit der Software zurecht zu kommen.		X ₁			

SUS - Score: $37 : 25 = \underline{92,5}$

Feedback des Probanden

Haben Sie noch allgemeine Anmerkungen zu diesem Test?

- zeitliche Verzögerung war problematisch
- hat aufgegriffen, was sie aus anderen Anwendungen kennt

Wie würden Sie das Ihnen vorliegende Programm verbessern?

✓

Vielen Dank für Ihre Teilnahme.

Eigenständigkeitserklärung

Ich versichere, dass ich die vorgelegte Bachelorarbeit eigenständig und ohne fremde Hilfe verfasst, keine anderen als die angegebenen Quellen verwendet habe und die den benutzten Quellen entnommenen Passagen als solche kenntlich gemacht habe.

Patrick Creutzburg

Stadtilm, den 7. April 2021