

# **Visualisierung einfacher Algorithmen für den Informatikunterricht**

Diplomarbeit an der  
Technischen Universität Dresden  
Januar 2008

Claudia Schindler

Betreuer: Dr. Holger Rohland  
Hochschullehrer: Prof. Dr. Steffen Friedrich  
Fachgebiet Didaktik der Informatik/Lehrerbildung  
Institut für Software- und Multimediatechnik  
Fakultät Informatik



# Erklärung

Hiermit erkläre ich, Claudia Schindler, die vorliegende Diplomarbeit zum Thema

**Visualisierung einfacher Algorithmen für den Informatikunterricht**

selbständig und ausschließlich unter Verwendung der im Quellenverzeichnis aufgeführten Literatur- und sonstigen Informationsquellen verfasst zu haben.

Dresden, am 30. Januar 2008

Unterschrift (Claudia Schindler)

## Aufgabenstellung für die Diplomarbeit

Name, Vorname des Studenten: Schindler, Claudia  
Immatrikulationsnummer: 2871696

Thema: "Visualisierung einfacher Algorithmen für den Informatikunterricht"

### Zielstellung:

Einfache Algorithmen bilden die Grundlage für das Programmieren. Obwohl die meisten Funktionen und Abfolgen logisch und trivial sind, bereiten sie den meisten Anfängern Schwierigkeiten. Ziel der Arbeit ist es deshalb zu untersuchen, wo die Ursachen für diese Schwierigkeiten liegen und wie diese Schwierigkeiten durch geeignete Visualisierung behoben werden können. Es sind mehrere solcher Visualisierungen zu erstellen und durch Einsatz in der Zielgruppe zu evaluieren.

Im Einzelnen werden folgende Ergebnisse angestrebt:

- Untersuchung aktueller Lösungen für die Visualisierung einfacher Algorithmen
- Analyse von wahrnehmungspsychologischen Aspekten hinsichtlich folgender Fragestellung:
  - Was sind die Ursachen für die Notwendigkeit einer Visualisierung für einfache Algorithmen?
  - Wo liegen die Schwierigkeiten im Verständnis und Verarbeiten?
  - Wie viele Informationen können in einer interaktiven Repräsentation wahrgenommen und vom Anwender verarbeitet werden?
  - Ist eine bestimmte Anordnung und Gestaltung der Elemente bzw. Teilvisualisierungen zu beachten?
  - Wie ist eine Visualisierung optimal zu gestalten?
- Erstellung mehrerer web-basierter Visualisierungen unter Einsatz des Werkzeugs Flash
- Evaluierung der erstellten Tools durch Einsatz in der Zielgruppe
- Integration in bereits bestehende Website für Lehr- und Lernmaterial auf dem Sächsischen Bildungsserver

Betreuer: Dr. rer. nat. Holger Rohland  
Betreuender Hochschullehrer: Prof. Dr. paed. habil. Steffen Friedrich  
Institut: Software- und Multimediatechnik  
Beginn am: 01.06.2007  
Einzureichen am: 30.11.2007  
Unterschrift

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung .....</b>	<b>1</b>
1.1	Zielsetzung .....	1
1.2	Aufbau der Arbeit .....	2
<b>2</b>	<b>Psychologische Aspekte der Kognition.....</b>	<b>4</b>
2.1	Gedächtnis und Aufmerksamkeit .....	4
	Gedächtnismodell.....	4
	Das visuelle Gedächtnis .....	6
	Visuelle Aufmerksamkeit .....	7
	Selektive visuelle Aufmerksamkeit .....	7
	Kapazität der visuellen Aufmerksamkeit .....	8
2.2	Visuelle Prozesse .....	10
	Visuelle Wahrnehmung .....	10
2.3	Visuelle Repräsentationen – Theorien und Modelle .....	13
	Dual-Coding Theorie (Paivio) .....	13
	Multimodale Gedächtnistheorie (Engelkamp) .....	14
2.4	Kognitive Modelle multimedialen Lernens.....	17
	Cognitive Theory of Multimedia Learning (Mayer) .....	17
	Integratives Modell des Text- und Bildverstehens (Schnotz & Bannert).....	18
	Wertung.....	19
	Lernen mit dynamisch-interaktiven Bildern .....	19
	Cognitive Load Theory (Sweller) .....	20
	Intensität des Bildverstehens .....	22
2.5	Wissenserwerb mit Bildern.....	22
	Lernleistungen .....	22
	Kategorisierung von Bildern .....	23
	Modelle der Bildverarbeitung.....	23
	Literacy .....	24
2.6	Mentale Modelle .....	26
	Abstraktion .....	27
	Mentale Modelle .....	27
	Bilder und mentale Modelle .....	29
	Bilder als Material für mentale Modelle.....	29
	Misskonzepte/Fehlvorstellungen .....	30
2.7	Zusammenfassung .....	31
	Notwendigkeit der Visualisierung einfacher Algorithmen.....	32

<b>3</b>	<b>Informatikunterricht .....</b>	<b>34</b>
3.1	Allgemeine Situation im Unterricht .....	34
	Informatikunterricht ist anders – Beobachtungen aus dem Unterricht .....	34
	Schwierigkeiten der Schüler .....	35
	Schwierigkeiten der Lehrer .....	36
3.2	Verbesserungsvorschläge .....	36
3.3	Fazit.....	39
<b>4</b>	<b>Analyse aktueller Werkzeuge .....</b>	<b>40</b>
4.1	DIN EN ISO 9241 .....	40
	Grundsätze der Dialoggestaltung – ISO 9241 Teil 110.....	40
	Anforderungen an die Gebrauchstauglichkeit – ISO 9241 Teil 11 .....	43
4.2	Bewertungskriterien .....	43
4.3	Werkzeuge .....	45
	j-Algo.....	45
	go2algo .....	46
	Algorithms in Action (AIA) .....	48
	Sortieren 2 .....	50
	AlgAE – Algorithm Animation Engine .....	51
	ANIMAL.....	52
	Jeliot 3.....	52
	ALVIS Live! .....	54
	Wertung der Werkzeuge .....	56
4.4	Fazit.....	57
<b>5</b>	<b>Gestaltungshinweise .....</b>	<b>58</b>
5.1	Allgemeine Gestaltungsprinzipien .....	58
	Gestaltgesetze .....	59
	Interaktion .....	60
5.2	Spezielle Gestaltungshinweise.....	61
	Gestaltung von Texten.....	61
	Gestaltung von Bildern .....	62
	Gestaltung von Animationen .....	64
5.3	Zusammenfassung .....	65

<b>6</b>	<b>Eigene Visualisierung.....</b>	<b>66</b>
6.1	Ideensammlung .....	66
	Anpassung des Lehrmaterials .....	67
6.2	Konzeption – Von der Realität zur Abstraktion .....	67
	Abstraktionsprozess .....	68
	Verstehen und Begreifen .....	70
6.3	Prototyp und dessen visuelle Elemente .....	71
	Programmbeispiel.....	73
6.4	Module im Prototyp.....	74
	Zuweisungen.....	74
	Boolesche Ausdrücke.....	75
	Schleifen (Kontrollstrukturen) .....	75
	Verzweigungen (Kontrollstrukturen) .....	77
6.5	Interna des Prototyps .....	79
	Aufbau .....	80
	Erweiterung um eigene Module .....	82
	Aktuelle Einschränkungen .....	82
<b>7</b>	<b>Evaluierung und Veröffentlichung .....</b>	<b>83</b>
7.1	Auswertung des Fragebogens.....	83
	Aussehen .....	83
	Inhalt.....	84
	Bedienbarkeit .....	85
	Wirkung .....	85
	Allgemeines .....	85
	Fazit .....	86
7.2	Veröffentlichung.....	86
<b>8</b>	<b>Zusammenfassung.....</b>	<b>87</b>
<b>9</b>	<b>Ausblick.....</b>	<b>90</b>
	<b>Anhang A – Werkzeugbewertung.....</b>	<b>92</b>
	<b>Anhang B – Evaluierung .....</b>	<b>110</b>
	<b>Abbildungsverzeichnis .....</b>	<b>126</b>
	<b>Tabellenverzeichnis .....</b>	<b>128</b>
	<b>Literaturverzeichnis .....</b>	<b>129</b>
	<b>Webverzeichnis.....</b>	<b>132</b>

# 1 Einleitung

Wenn man eine Fremdsprache erlernt, müssen Vokabeln und die Grammatik der Sprache Schritt für Schritt erworben werden. Unterhält man sich nun mit einem Muttersprachler, wird häufig das Defizit des eigenen erlernten Vokabulars und der Ausdrucksweise deutlich. Dennoch erlaubt die natürliche Integration von Mimik und Gestik eine erfolgreiche Kommunikation. Trotz möglicher grammatikalischer Fehler, der Verwendung falscher Worte oder der Einsatz von Bruchstücken, wird der Gegenüber den Inhalt verstehen.

Programmieren zu lernen ist ähnlich. Eine Programmiersprache besteht wie jede andere Sprache aus einem Wortschatz und einer Grammatik. Ihr Wortschatz ist stark eingeschränkt und auf ein Minimum reduziert. Zudem ist die Auswirkung der Korrektheit einer Programmiersprache nicht mit der einer natürlichen Sprache vergleichbar. Bereits kleinste Fehler führen zu einem Unverständnis beim Kommunikationspartner. Das eigentliche Ziel der Sprache, die Übermittlung von Informationen, baut auf der Verwendung von Grammatik und Worten auf. Im Gegensatz zur natürlichen Sprache kann allerdings das eigene Anliegen nicht unter Verwendung der Körpersprache vermittelt werden. Die präzise Verwendung der syntaktischen Regeln und des gegebenen Vokabulars ist notwendig.

Doch wer die Syntax einer Programmiersprache kennt, muss nicht unbedingt deren Bedeutung verstanden haben. Schon bei den typischen Grundelementen der Sprache wie Wertzuweisungen (z.B.  $a := a + 1$ ) tritt fehlerhaftes Verständnis auf. Sie können durch die falsche Anwendung der syntaktischen Regeln hervorgerufen werden, aber auch durch eine mangelnde Vorstellung des Programmieranfängers entstehen. Dies tritt vor allem bei Programmieranfänger auf, für die es umso schwieriger ist, ihre Intension in die Programmiersprache zu transformieren und damit den Zusammenhang zu abstrahieren.

Programmieranfänger in dem Kontext dieser Arbeit sind Menschen, die bisher noch keine Berührungen mit dem Erstellen von Programmen hatten, zum Beispiel Schüler im Informatikunterricht.

## 1.1 Zielsetzung

Ziel der Arbeit ist eine Verbesserung der Vermittlung von Programmiergrundlagen mit spezieller Ausrichtung auf den Informatikunterricht. Als typische Grundelemente des Programmierens sind Wertzuweisungen, Schleifen und Verzweigungen zu sehen. Sie werden unter dem Begriff der einfachen Algorithmen zusammengefasst. Als Basis soll eine umfassende Ursachenforschung erfolgen, wie die Schwierigkeiten bei Programmieranfängern entstehen und sich festsetzen. Der zu analysierende Bereich wurde auf die psychologischen Aspekte der Kognition und die Situation im Informatikunterricht fixiert.

Der Mensch speichert Informationen in einem visuellen und verbalen Kontext, welcher in Form einer mentalen Repräsentation jederzeit abrufbar ist. Unter der Berücksichtigung des Umgangs mit visuellen Präsentationen und ihrer Verarbeitung im Gehirn, sowie der Analyse aktueller Visualisierungen einfacher Algorithmen werden allgemeine Hinweise zur Gestaltung von Visualisierungen zusammengestellt. Mit dem Wissen der psychologischen Kognition und den Gestaltungshinweisen wird ein Konzept „Von der Realität zur Abstraktion“ erstellt und anschließend als ein Prototyp umgesetzt. Das erstellte Werkzeug ist für den unterrichtsbegleitenden Einsatz gedacht.

## 1.2 Aufbau der Arbeit

Um diese Ziele zu erreichen, wird zunächst im Kapitel 2 ein Überblick zum menschlichen Gedächtnis, der Aufmerksamkeit und der Wahrnehmung, als Grundlage für die psychologischen Aspekte der Kognition, geschaffen. Das Spektrum der Kognition reicht von der Wahrnehmung und dem Erkennen von Objekten aus der Umwelt über das Denken, Erinnern und Lernen bis hin zu Erwartungen und Problemlösungen. Im Vordergrund stehen die visuellen Prozesse der Wissensverarbeitung und -speicherung, die anhand verschiedener Modelle veranschaulicht werden. Dabei haben alle vorgestellten Modelle eines gemeinsam, sie werden mit Hilfe von verbalen und visuellen Elementen im Zuge eines Abstraktionsprozesses erstellt. Menschen versuchen solche mentale Darstellungen in die Realität, Form von Bildern zurückzuübertragen, wobei immer wieder Schwierigkeiten auftreten, sodass auch der Umgang mit bildhaften Darstellungen in der Ursachenforschung berücksichtigt wird.

Eine zentrale Position nimmt die Lernumgebung des Anfängers ein, die er im Informatikunterricht erfährt. Dazu wird im Kapitel 3 die Situation im Informatikunterricht analysiert. Fragen, durch welche Verhaltensmuster sich Lehrer und Schüler in diesen Unterrichtsstunden auszeichnen und wie sich die Lernsituation entsprechend gestaltet, sollen beantwortet werden.

Ein Schwerpunkt der Arbeit ist die Analyse aktueller Visualisierungswerkzeuge, deren Fokus auf einfachen Algorithmen und dem unterrichtsbegleitenden Einsatz liegt. Zur Bewertung der Anwendungen wird die ISO-Norm 9241 (*Ergonomie der Mensch-System-Interaktion*) genutzt, welche die Richtlinien der Interaktion zwischen Mensch und Computer beschreibt. Sie dient als Grundlage für ein Bewertungsraster, anhand dessen die Analyse in Kapitel 4 durchgeführt wird.

Die bisherigen Kapitel enthalten die Forschung nach Ursachen für die Schwierigkeiten von Programmieranfängern und die Wertung aktueller Visualisierungen einfacher Algorithmen. Im Kapitel 5 werden Erkenntnisse zur Verbesserung der grafischen Gestaltung zusammengetragen. Sie werden in Form von Hinweisen für die Gestaltung von Elementen und dem Gesamtkonzept empfohlen.



Im Mittelpunkt der Arbeit steht die Entwicklung eines eigenen Werkzeugs zur Visualisierung einfacher Algorithmen. Das zugrunde liegende Konzept „Von der Realität zur Abstraktion“ soll Programmieranfänger bei ihren ersten Schritten unterstützen. Dessen Herleitung und prototypische Umsetzung werden in Kapitel 6 dargelegt.

Das Kapitel 7 liefert erste Impressionen der Praxistauglichkeit des Werkzeugs. Dazu wurden im Zuge einer Evaluierung Lehrer und Studenten gebeten das Werkzeug zu testen, zu bewerten und zu kommentieren. Eine entsprechende Auswertung ist Bestandteil des Kapitels.

Den Abschluss bilden die Zusammenfassung der Arbeit (Kapitel 8) und ein Ausblick auf die weiterführende Entwicklung und Erweiterung des Werkzeugs (Kapitel 9).

## 8 Zusammenfassung

Ziel der Arbeit war es eine Verbesserung der Vermittlung von Programmiergrundlagen mit spezieller Ausrichtung auf den Informatikunterricht zu schaffen. Als typische Grundelemente sind Wertzuweisungen, Schleifen und Verzweigungen zu sehen und wurden unter dem Begriff einfache Algorithmen zusammengefasst. Es sollte eine umfassende Ursachenforschung durchgeführt werden, welche die Schwierigkeiten von Programmieranfängern aufzeigt. Der Bereich psychologischer, kognitiver Prozesse sowie die Situation im Informatikunterricht waren Gegenstand der Analyse.

Der Einstieg in die Psychologie der visuellen Wahrnehmung und der visuellen Prozesse im menschlichen Gehirn wurde mit einem kleinen Exkurs in die Gedächtnisstrukturen begonnen. Die Aufnahme visueller Reize über das Auge und deren Verarbeitung im Gehirn wurden kurz angerissen, um einen Bezug für die folgenden Theorien zur Text- und Bildverarbeitung zu gewähren. In den Theorien wird von einer dualen Aufbereitung der Information ausgegangen. Zwar werden die verbalen und non-verbalen Informationen getrennt voneinander verarbeitet, aber in der mentalen Repräsentation wieder miteinander verknüpft. Vorteil der dualen Kodierung ist die Existenz zweier Abbildungen unterschiedlicher Basis (verbal und imaginär). Die Sprache nimmt in diesen Modellen eine Sonderstellung ein, sie kann sowohl als verbale Information wie auch als non-verbale Information aufgenommen werden. Wird beispielsweise ein Wort gelesen, so kann es auch eine bildhafte Vorstellung erzeugen [ENG 91]. Texte, die über die visuelle Ebene aufgenommen werden, können aber auch in Laute konvertiert werden und so eine verbale Repräsentation erzeugt werden [MAY 01]. Die Kapazität für die Verarbeitung im Arbeitsgedächtnis ist beschränkt und auf verschiedene Elemente bezogen. So wird in der Cognitive Load Theory [SWE 94] von drei Belastungskomponenten (intrinsisch, lernirrelevant, lernrelevant) ausgegangen, welche die Verarbeitung beeinflussen. Sie sollten ausgewogen sein, sodass die Beschäftigung mit einem Typus sich nicht negativ auf die anderen auswirkt.

Eine weitere Erkenntnis war, dass Bilder im europäischen Kulturkreis als Informationsträger an zweiter Stelle stehen, an erster stehen Texte. Der Informationsgehalt in verbal beschriebenen Darstellungen wird höher gewertet als der in Bildern, obwohl Bilder einfache Hilfsmittel für die Aufbereitung von Informationen sind. Der Umgang mit ihnen sollte laut Petterson und Weidenmann [PET 94, WEI 94] stärker gefördert werden, denn schließlich sind sie auch die Grundlage für mentale Modelle.

In mentalen Modellen werden Sachverhalte aus der Realität vereinfacht und auf wesentliche Elemente reduziert, beschreiben Strukturen und Funktionsweisen der Sachverhalte. Dadurch entstehen abstrakte Repräsentationen, die jederzeit in dem menschlichen Gehirn abrufbar sind. Bei der Abstraktion können auch fehlerhafte Modelle entstehen, die als Fehlvorstellungen abgelegt werden. Diese Erkenntnis führte zu der Schlussfolgerung, dass Lernende beim Abstraktionsprozess unterstützt werden müssen.

Der nächste Punkt der Arbeit bildete die Untersuchung des Informatikunterrichts. Dazu wurden einzelne Probleme aufgezeigt, welche die Lernumgebung, die Gestaltung des Unterrichts (Stunde) sowie die Fähigkeiten von Lehrern und Schülern betreffen. Daraus ergab sich das Fazit, dass Informatikunterricht ganz anders ist [HUM 06a] und ganz andere Herangehensweisen benötigt. Der Unterricht benötigt eine Strukturierung, sodass alle Schüler diesen bewältigen können und die Besseren nicht unterfordert werden, eine Unterteilung in Pflicht- und Zusatzabschnitt des Lehrstoffs ist angebracht.

Ein weiterer Schwerpunkt lag auf der Bewertung aktueller Applikationen zur Thematik: Visualisierung einfacher Algorithmen, wobei zu betonen ist, dass es sich bei einfachen Algorithmen um Grundlagen des Programmierens handelt. Die Bewertungen basieren auf der ISO Norm 9241, welche die Richtlinien zur Interaktion von Mensch und Computer beschreibt. Von acht getesteten Werkzeugen erfüllt eigentlich keines das Kriterium einfacher Algorithmen. Als einfache Algorithmen werden meistens Sortier- und Suchalgorithmen, Bäume oder andere einfachere Probleme verstanden. Das Heranführen an die Programmierung wird in Werkzeugen wie Jeliot3 [JEL 07] und Alvis Live! [HUN 07] aufgegriffen, jedoch nur in Jeliot3 in einer ansprechenden und funktionalen Umgebung umgesetzt. Der Lernende bekommt seinen Programmcode (Java) in einer anschaulichen, aber nah an der Programmierenebene orientierten Visualisierung veranschaulicht und kann den Ablauf auf der Bühne verfolgen. Ein weiteres Werkzeug mit einem interessanten Konzept ist ANIMAL [ANI 07]. Für dieses wird eine Vielzahl von kleinen Modulen auf der Homepage zur Verfügung gestellt. Eine einfache Integration der einzelnen Module zeichnet dieses Werkzeug aus.

Aus den Erkenntnissen der Kognition und visueller Verarbeitung von verbalen und non-verbalen Elementen sowie der Situation im Unterricht und den zuvor betrachteten Werkzeugen, sollten Gestaltungshinweise, Designprinzipien und Hinweise für die optimale Gestaltung abgeleitet werden. Wichtig sind Texte, Bilder, Animationen bzw. Filme. Der Europäer ist textfixiert und braucht stets einen verbalen Bezug zu Bildern und Animationen. Eine Referenz im Text und eine Anleitung zum „Lesen“ des Bildes sind notwendig. Da Lernwerkzeuge meist am Computer präsentiert werden, lassen sich einige Grundlagen für deren Gestaltung aus dem Webdesign anwenden.

Trotz der Textfixierung sind Bilder der Schlüssel zum menschlichen Gedächtnis, denn im Gedächtnis werden viele Ereignisse in Form von mentalen Repräsentationen abgelegt. Anforderungen an deren Gestaltung sind präzise, korrekte und einfache Darstellung, welche mit den notwendigen gestalterischen Elementen versehen werden sollten. Animationen sind als Fortführung von Bildern zu sehen, für sie gelten die gleichen Bedingungen, müssen aber noch ergänzt werden. Sie sind Sequenzen von Bildern und werden in einem kleineren Zeitfenster präsentiert und wahrgenommen, deswegen müssen sie weniger komplex und mit besonderem Augenmerk auf das wesentliche gestaltet sein.

Mit den erarbeiteten Grundlagen, wurde eine eigene Applikation zur Visualisierung einfacher Algorithmen entwickelt, welche auf dem Konzept „Von der Realität zur Ab-

straktion“ basiert. Immer wieder scheitern Schüler an den Schwierigkeiten, die durch mangelndes Abstraktionsvermögen entstehen. Mit Hilfe von Analogien aus dem Alltag soll der Abstraktionsprozess erleichtert werden. Ein Beispiel aus der Realität wird in mehreren Reduktionschritten in eine abstrakte Schreibweise übertragen, den Quellcode. Die Idee ist, den Prozess, wie er bei der Erstellung von mentalen Repräsentationen eingesetzt wird, zu verwenden. Dies erfolgt durch die Beschränkung auf das Wesentliche, das Weglassen von Unwesentlichem und dem Vereinfachen. Die Reduktion basiert auf dem Vergleich von natürlichen Sprachen und Programmiersprachen. Beide Sprachtypen basieren auf einem Wortschatz und einer dazugehörigen Grammatik. Jedoch besitzen Programmiersprachen nur einen minimalen Wortschatz und eine sehr abstrakte Grammatik.

Die Transformation in den Quellcode bildet die letzte Phase, nun geht es darum dem Lernenden den Inhalt der Codezeilen begreiflich zu machen. Dies passiert im Werkzeug auf der Basis eines Quellcodes in dem die aktuellen Wertbelegungen der Variablen berücksichtigt und ausgewertet werden. Das Programmbeispiel wird als animierte Visualisierung parallel in einem weiteren Fenster angezeigt. Ein separates Belegungsprotokoll kann anstelle der Animation eingeblendet werden. Diese letzte Phase wurde prototypisch umgesetzt.

In der abschließenden Evaluierung wurde der Prototyp von Lehrern bzw. angehenden Lehrern für das Fach Informatik getestet und bewertet. An der Evaluierung nahmen 14,2 Prozent teil. Das Werkzeug wurde durchschnittlich mit der Note 1,82 bewertet, welches der Schulnote Gut entspricht. Es würde von etwa der Hälfte im Unterricht eingesetzt werden. Das Werkzeug wurde anderen Lehrern auf der Plattform des sächsischen Bildungsservers zur Verfügung gestellt.

## 9 Ausblick

Probleme beim Verständnis und Vorstellungen diverser Sachverhalte aus dem Bereich Computer und Technik wird es immer geben, aber in Zukunft müssen sie besser bewältigt werden. Es ist notwendig den Lernenden mit interaktiven und multimedialen Anwendungen zu unterstützen, Medien und Grundlagen sind schließlich vorhanden. Sie müssen besser genutzt und in die Materialien integriert werden.

Für das Erstellen solcher Anwendungen müssen psychologische Verarbeitungsprozesse im Umgang mit dynamischen und interaktiven Bildern/Situationen weiter erforscht werden. Auf diesen Grundlagen können spezifische Richtlinien (Styleguides) zur Gestaltung von multimedialen Anwendungen und speziell interaktiven Lernmedien erarbeitet werden.

Das entwickelte Konzept „Von der Realität zur Abstraktion“ sollte weiter ausgebaut und umgesetzt werden, denn bisher wurde nur die letzte Phase als Prototyp gestaltet. In dem Konzept steckt Potenzial dem Programmieranfänger beim Erlernen des Programmierens, speziell beim Verstehen und Begreifen, zu helfen. Der schwierige Weg bei einem Sachverhalt die wesentlichen Merkmale zu erkennen und den Prozess zu abstrahieren, könnte damit erleichtert werden.

Im Werkzeug sind bereits acht Module mit Programmiergrundlagen integriert, weitere Module können ergänzt werden. Dabei sollte aber immer die Kernidee, einfache Algorithmen zu visualisieren, berücksichtigt werden. Es sei noch einmal auf die Wichtigkeit elementarer Grundlagen hingewiesen. Daraus ergeben sich Themen wie Prozeduren, Funktionsaufrufe und Rekursion. Diese fordern eine Erweiterung des Werkzeugs um die Funktionalität eines Stapelspeichers (Stack).

Aktuell beschränkt sich die Interaktivität in den Komponenten auf ein Minimum, der Lernende wird an diversen Stellen zur Eingabe von Werten aufgefordert und muss den Ablauf steuern. Wichtig wäre die Manipulation von Werten in der Animation und den dazugehörigen Auswirkungen auf den Quellcode, beispielsweise das Verschieben von Schwellwerten in den Modulen zu den Verzweigungen, zu ermöglichen.

Der Lernende bekommt bereits Informationen zum Modul angeboten, weitere Ansichten z.B. Struktogramme wären denkbar. Struktogramme helfen den Lernenden den Algorithmus aufzustellen und in den spezifischen Quellcode zu übertragen. Es wäre möglich sie in einem weiteren Fenster, wie Animation und Belegungsprotokoll oder in Form eines Dialoges wie die Informationen zum Modul zu integrieren.

Die Pseudosprache im Quellcode könnte flexibel und austauschbar gestaltet werden, sodass auch konkrete Programmiersprachen wie Pascal oder Java bzw. auch eine Vereinfachung mit deutschen Schlüsselwörtern angezeigt werden könnte. Dies würde den Einsatz für Lernende ohne Fremdsprachenkenntnisse ermöglichen aber auch die Anwendung für spezielle Sprachen fördern.

Zudem gab es Funktionalitäten, die aufgrund des zeitlichen Rahmens dieser Arbeit nicht implementiert wurden. Dies betrifft zum Beispiel das schrittweise Zurückspringen sowie die individuelle Anpassung der Abspielgeschwindigkeit.

Alle Erweiterungen sollten in einem größeren Rahmen und mit den entsprechenden Zielgruppen evaluiert werden.

# Abbildungsverzeichnis

Abbildung 1: Integriertes Gedächtnismodell.....	6
Abbildung 2: Modell des Arbeitsgedächtnisses nach Baddeley (nach [BAD 02]) .....	9
Abbildung 3: Visuelle Prozesse im Gehirn.....	10
Abbildung 4: Querschnitt des menschlichen Auges [OPT 07].....	11
Abbildung 5: Das menschliche Gehirn von der Seite. ....	12
Abbildung 6: Die Dual-Coding-Theory von Paivio .....	13
Abbildung 7: Multimodale Gedächtnistheorie von Engelkamp (nach [ENG 91]) .....	15
Abbildung 8: Struktur des verbalen sensomotorischen Systems nach Engelkamp .....	16
Abbildung 9: Cognitive Theory of Multimedia Learning (nach [MAY 01]) .....	17
Abbildung 10: Integratives Modell des Text- und Bildverstehens [SCH 03, SCH 02] .....	19
Abbildung 11: Begrenzte Kapazitäten des Arbeitsgedächtnisses.. ....	21
Abbildung 12: Entstehen einer mentalen Abbildung. ....	28
Abbildung 13: Aktivitäten zur Behaltensleistung [SNO 02] .....	35
Abbildung 14: J-algo - Screenshot aus dem Modul des Knuth-Morris-Pratt-Algorithmus mit der Suche des Musters in einem Kontext.....	46
Abbildung 15: go2algo - Bubble-Sort. ....	48
Abbildung 16: QuickSort in AIA. ....	49
Abbildung 17: Sotieren 2. Visualisierung des Bubble-Sort Algorithmus.....	50
Abbildung 18: AlgAE V2.0 in Aktion. ....	51
Abbildung 19: ANIMAL Visualisierung am Beispiel von HeapSort. ....	52
Abbildung 20: Screenshot Jeliot3.....	54
Abbildung 21: Alvis Live! .....	55
Abbildung 22: Ablaufmodell zum Design von Bildern (nach Rankin [WEI 94]).....	64
Abbildung 23: Konzeption „Von der Realität zur Abstraktion“ mit zwei Handlungsebenen „Abstraktionsprozess“ und „Verstehen & Begreifen“.....	68
Abbildung 24: Nachbildung des Transformationsschrittes .....	69
Abbildung 25: Ebene des Abstraktionsprozesses im Konzept .....	70
Abbildung 26: Integriertes Belegungsprotokoll .....	71
Abbildung 27: Konzept im Überblick.....	71
Abbildung 28: Quellcode mit Animation.....	72
Abbildung 29: Quellcode mit Belegungsprotokoll. ....	73
Abbildung 30: Screenshot aus dem Modul .....	75

---

Abbildung 31: Metapher Wasserglas .....	76
Abbildung 32: Screenshot der Metapher Thermometer .....	78
Abbildung 33: Ausschnitt aus dem Klassendiagramm .....	81



# Tabellenverzeichnis

Tabelle 1: Bilder im Schulalltag .....	25
Tabelle 2: Aspekte der Infologie [PET 94] .....	26
Tabelle 3: Verschiedene Suchrichtungen zum Zusammenhang von Bildern und mentalen Modellen [WEI 94] .....	29
Tabelle 4: Bewertungsraster für Visualisierung von Algorithmen .....	44
Tabelle 5: Wichtigste Gestaltgesetze und ihre Erläuterungen [EYS 00, IWG 07].....	60
Tabelle 6: Ideen für den Einsatz von Visualisierungen.....	66
Tabelle 7: Umsetzungsideen .....	67
Tabelle 8: Arten von Schleifen .....	76
Tabelle 9: Drei Arten der Verzweigung .....	77

# Literaturverzeichnis

- [BAD 02] Baddeley, A. D. (2002). Is working memory still working? *European Psychologist*, 7, S. 85-97.
- [BED 06] Bednarik, R., Moreno, A. & Myller, N. (2006). Program Visualization for Programming Education – Case of Jeliot 3. *ACM NZ Bulletin*, 2(2), ISSN 1176-9998, 2006.
- [BOD 04] Bodemer, D. (2004). Unterstützung multimedialen Lernens. *Dissertation*. Freiburg.
- [CEN 95] Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten. (Februar 1995). Teil 10: Grundsätze der Dialoggestaltung (ISO 9241-10 : 1995). Brüssel: CEN (Europäisches Komitee für Normung).
- [ENG 91] Engelkamp, J. (1991). *Das menschliche Gedächtnis*. Göttingen: Verlag für Psychologie Dr. C. J. Hogrefe. ISBN 3-8017-0344-4.
- [EUL 92] Euler, D. (1992). *Didaktik des computerunterstützten Lernens: praktische Gestaltung und theoretische Grundlagen*. Nürnberg: BW Bildung und Wissen Verlag und Software GmbH. ISBN 3-8214-7014-3.
- [EYS 00] Eysenck, M. W. (2000). *Psychology. A Student's Handbook*. Hove/East Sussex: Psychology Press Ltd. ISBN 0-86377-551-9.
- [GÜR 05] Gürth, R. (2005). Sortieren 2. *Seminararbeit*. Technische Universität Dresden.
- [HAR 06] Hartmann, W., Näf, M., & Reichert, R. (2006). *Informatikunterricht planen und durchführen*. Berlin, Heidelberg: Springer-Verlag. ISBN 978-3-540-34484-1.
- [HAS 06] Hasselhorn, M., & Gold, A. (2006). *Pädagogische Psychologie*. Stuttgart: W. Kohlhammer. ISBN 978-3-17-017110-7.
- [HUB 04] Hubwieser, P. (2004). *Didaktik der Informatik – Grundlagen, Konzepte, Beispiele*. Berlin, Heidelberg: Springer-Verlag. ISBN 3-540-43510-7.
- [HUM 06a] Humbert, L. (2006). *Didaktik der Informatik mit praxiserprobtem Unterrichtsmaterial*. Wiesbaden: B.G. Teubner Verlag / GWV Fachverlage GmbH. ISBN 978-3-8351-0112-8.
- [KAI 99] Kaiser, J. (1999). *Bewertung multimedialer Darstellungsformen unter didaktischen Gesichtspunkten*. Diplomarbeit. Paderborn: Universität-GH Paderborn
- [KRA 05] Kramer, B. (2005). *Mentale Integration von Text und Bildern beim Lernen mit Multimedia*. Dissertation. Kiel: Christian-Albrechts-Universität zu Kiel.
- [MAY 01] Mayer, R. E., Heiser, J., Lonn, S. (2001). Cognitive Constraints on Multimedia Learning: When Presenting More Material Results in Less Understanding. *Journal of Educational Psychology*, Vol. 93, No. 1, S. 187-198, DOI: 10.1037//0022-0663.93.1.187.
- [MER 83] Merrill, M. D. (1983). Component Display Theory. In C. M. Reigeluth, *Instructional-design theories and models. An overview of their current status* (S. 284-334). Hillsdale: Lawrence Erlbaum. ISBN 0-89859-275-5.

- [MÜS 02] Müsseler, J., Prinz, W. (2002). *Allgemeine Psychologie*. Heidelberg, Berlin: Spektrum Verlag. ISBN 3-8274-1128-9.
- [PAL 99] Palmer, S. E. (1999). *Vision Science – Photons to Phenomenology*. Cambridge, Massachusetts: MIT Press. ISBN 0-262-16183-4.
- [PAI 90] Paivio, A. (1990). *Mental Representations: A Dual Coding Approach*. New York: Oxford University Press US. ISBN 978-0195066661.
- [PET 94] Petterson, R. (1994). Visual Literacy und Infologie. In B. Weidenmann, *Wissens-erwerb mit Bildern: instruktionale Bilder in Printmedien, Film/Video und Computer-programmen* (S. 215 – 235). Bern: Verlag Hans Huber. ISBN 3-456-82415-7.
- [SAL 84] Salomon, G. (1984). Television is „easy“ and print is „tough“: The differential investment of mental effort in learning as a function of perceptions and attributions. *Journal of Educational Psychology*, 76, 1984, S. 647-658
- [SNO 94] Schnotz, W. (1994). Wissenserwerb mit logischen Bildern. In B. Weidenmann, *Wissenserwerb mit Bildern* (S. 95-147). Bern: Verlag Hans Huber. ISBN 3-456-82415-7.
- [SNO 02] Schnotz, W. (2002). Wissenserwerb mit Texten, Bildern und Diagrammen. In L. J. Issing, & P. Klimsa, *Information und Lernen mit Multimedia und Internet* (S. 65-81). Weinheim: Verlagsgruppe Beltz, Psychologie Verlags Union. ISBN 3-621-27449-9.
- [SNO 03] Schnotz, W., & Bannert, M. (2003). Construction and interference in learning from multiple representations. *Learning and Instruction*, 13(2), S. 141-156.
- [SUL 07] Schulmeister, R. (2007). *Grundlagen hypermedialer Lernsysteme. Theorie, Didaktik, Design*. München: Oldenbourg. ISBN 978-3-486-27395-3.
- [STA 97] Stary, J. (1997). *Visualisieren: ein Studien- und Praxisbuch*. Berlin: Cornelsen Scriptor. ISBN 3-589-21077-X.
- [STE 94] Stebler, R., Reusser, K. & Pauli, C. (1994). Interaktive Lehr-Lern-Umgebungen: Didaktische Arrangements im Dienste des gründlichen Verstehens. In K. Reusser & M. Reusser-Weyeneth, *Verstehen: psychologischer Prozess und didaktische Aufgabe*. Bern: Verlag Hans Huber. ISBN 3-456-82441-6.
- [SWE 88] Sweller, J. (1988). Cognitive Load During Problem Solving: Effects on Learning. *Cognitive Science*, 12(2), S. 257-285.
- [SWE 94] Sweller, J. & Chandler, P. (1994). Why Some Material Is Difficult to Learn. *Cognition and Instruction*, 12(3), S. 185-233.
- [TIE 03] Tietze, H. (2003). *Die neuronale Organisation des konzeptuellen Wissens*. Dissertation. Jena: Friedrich Schiller Universität Jena.
- [WIE 94] Weidenmann, B. (1994). Informierende Bilder. In B. Weidenmann, *Wissenserwerb mit Bildern: instruktionale Bilder in Printmedien, Film/Video und Computerpro-grammen* (S. 9 – 58). Bern: Verlag Hans Huber. ISBN 3-456-82415-7.

- [ZEL 89] Zeller, D. (1989). *Didaktische und methodische Überlegungen zu Sortierverfahren*. Freie und Hansestadt Hamburg: Behörde für Schule, Jugend und Berufsbildung, Amt für Schule.
- [ZIM 04] Zimbardo, P. G. & Gerrig, R. J. (2004). *Psychologie*, 16. Aktualisierte Auflage. München: Pearson Studium. ISBN 3-8273-7056-6.

# Webverzeichnis

- [ADO 08] *Adobe Flex 2 – SDK*. (kein Datum). Abgerufen am 23. Januar 2008 von <http://www.adobe.com/de/products/flex/sdk/>
- [ANI 07] *ANIMAL-Homepage*. (kein Datum). Abgerufen am 25. November 2007 von <http://www.animal.ahrgr.de/de/description.php3>
- [BIN 05] *binaervarianz.de – j-algo*. (16. Januar 2005). Abgerufen am 20. August 2007 von [binaervarianz.de: http://j-algo.binaervarianz.de/index.php](http://j-algo.binaervarianz.de/index.php)
- [BLU 98] Blumstengel, A. (1998). *Entwicklung hypermedialer Lernsysteme*. Abgerufen am 14. September 2007 von [http://www.dsor.upb.de/~blumstengel/main\\_index\\_titel.html](http://www.dsor.upb.de/~blumstengel/main_index_titel.html)
- [DON 04] Donker, H. (2004). Gestaltung von Groupware. Technische Universität Dresden. Abgerufen am 23. Januar 2008 von [http://web.inf.tu-dresden.de/KMA/downloads/SomSem\\_04/VL\\_Groupware/PPT\\_VL\\_Metaphern.pdf](http://web.inf.tu-dresden.de/KMA/downloads/SomSem_04/VL_Groupware/PPT_VL_Metaphern.pdf)
- [ECL 08] *Eclipse Framework*. (kein Datum). Abgerufen am 23. Januar 2008 von <http://www.eclipse.org/>
- [FDT 08] FDT Plugin. (kein Datum). Abgerufen am 23. Januar 2008 von <http://fdt.powerflasher.com/>
- [GEG 07] Gegenfurtner, K. R., Walter, S. & Braun, D. I. (kein Datum). *Allgemeine Psychologie Universität Gießen*. Abgerufen am 14. September 2007 von Visuelle Informationsverarbeitung im Gehirn: <http://www.allpsych.uni-giessen.de/karl/each/aka.html>
- [GO2 04] *go2algo*. (2004). Information. Abgerufen am 10. September 2007 von <http://www.syssec.at/~peter/docs/go2algo.pdf>
- [HUM 06b] Humbert, L. (2006). *Hammer Schulen im Netz e.V.* Abgerufen am 12. September 2007 von Informatische Bildung: Fehlvorstellungen und Standards: [http://www.ham.nw.schule.de/pub/bscw.cgi/d291985/2006-05-05\\_MWS-Fehlvorstellung-Standards.pdf](http://www.ham.nw.schule.de/pub/bscw.cgi/d291985/2006-05-05_MWS-Fehlvorstellung-Standards.pdf)
- [HUN 07] Hundhausen, C. (2007). *Alvis Live!* Abgerufen am 20. Oktober 2007 von VEUPL: <http://eecs.wsu.edu/~veupl/soft/alvis/index.htm>
- [IWG 07] IWM – Institut für Wissensmedien. (9. August 2007). *Gestaltgesetze*. Abgerufen am 7. Oktober 2007 von [e-teaching.org: http://www.e-teaching.org/didaktik/gestaltung/visualisierung/gestaltgesetze/](http://www.e-teaching.org/didaktik/gestaltung/visualisierung/gestaltgesetze/)
- [IWS 06] IWM – Institut für Wissensmedien. (17. Februar 2006). *Screendesign*. Abgerufen am 6. Oktober 2007 von [e-teaching.org: http://www.e-teaching.org/didaktik/gestaltung/screendesign/](http://www.e-teaching.org/didaktik/gestaltung/screendesign/)
- [JEL 07] *Jeliot*. (kein Datum). Abgerufen am 16. Oktober 2007 von Joensuu yliopisto – Tietojenkäsittelytiede: <http://cs.joensuu.fi/~jeliot/index.php>
- [OPT 07] *Optik Knipprath, Würselen, das Auge*. (kein Datum). Abgerufen am 27. September 2007 von Optik Knipprath: <http://www.optikknipprath.de/article5305-1538.html>
- [SBS 08] *PäPIKK – Pädagogische Plattform – Interaktion – Kommunikation – Kooperation*. (kein Datum). Abgerufen am 28. Januar 2008 von <http://www.sn.schule.de/>

- [SLE 06] Schlemm, A. (2006). *open theory - offene theorie: Mentale Modelle - ein zweiter Beitrag*. Abgerufen am 24. August 2007 von [www.opentheory.org/kf\\_mentales\\_modell2/text.phtml](http://www.opentheory.org/kf_mentales_modell2/text.phtml)
- [SUH 04] Schuhmacher, J. (2004). *Controlling 21*. Abgerufen am 22. September 2007 von <http://www.multimedia-beratung.de/ergonomie/theorie/grundlagen/grundlagen.htm>
- [SUL 05] Schulmeister, R. (2005). *Interaktivität in Multimedia-Anwendungen*. Abgerufen am 1. Oktober 2007 von e-teaching.org: <http://www.e-teaching.org/didaktik/gestaltung/interaktiv/InteraktivitaetSchulmeister.pdf>
- [STE 02] Stern, L., Naish, L., & Sondergaard, H. (2002). *Algorithms in Action*. Abgerufen am 6. September 2007 von <http://www.cs.mu.oz.au/aia>
- [ZEI 99] Zeil, D. S. (1999). *AlgAE - Algorithm Animation Engine*. Abgerufen am 29. September 2007 von AlgAE - Algorithm Animation Engine: [www.cs.odu.edu/~zeil/algae.html](http://www.cs.odu.edu/~zeil/algae.html)