



*„Le Corbusier, skizzierend“*  
Foto: René Burri – Magnum

## **„Ein parametrischer Ansatz für die Echtzeit-Interpretation von Handskizzen in den frühen Phasen des Architekturentwurfs“**

Diplomarbeit am Lehrstuhl für Informatik in der Architektur

Bauhaus-Universität Weimar

Autor: Jakob Beetz

Betreuung durch:

Prof. Dr. Dirk Donath

Zweitgutachter

Ju.-Prof. Dr. Frank Petzold

*„Nulla dies sine linea! [...] Die Skizze ist die erste und einfachste Art des Zeichnens, die den Gebrauch der Linie in ihrer reinsten Form zeigt und meist dem Umrisszeichnen gleichgesetzt wird. In der Umrisszeichnung sind die verschiedenen Aspekte eines Objekts (einer Architektur) ohne die Wirkung von Licht und Schatten auf ein vereinfachtes ideales Ergebnis beschränkt. Skizzieren ist eine Art Nachdenken auf dem Papier. Die Sprache des Architekten ist also die Zeichnung. Man kann die Architekturzeichnung als dokumentierte Struktur ansehen, in der die Linie oder andere Chiffren nach dem Willen des Zeichners Zusammenhänge sichtbar machen.“*

Gustav Peichl: „Die Sprache des Architekten  
ist die Zeichnung“

## Inhalt

|  |    |
|--|----|
| I Einleitung.....  | 5  |
| 1 Gliederung der Arbeit .....                                      | 6  |
| II Computergestütztes Skizzieren im architektonischen Kontext..... | 8  |
| 1 Die Skizze im architektonischen Kontext.....                     | 8  |
| 1.1 Kurzer Geschichtlicher Abriß.....                              | 8  |
| 2 Modelle zur computergestützten Beschreibung von Architektur..... | 8  |
| 2.1 Der Begriff des Modells in der Bauwerksbeschreibung.....       | 9  |
| 2.2 Etablierte Standards zur Beschreibung von Gebäudemodellen..... | 10 |
| 2.2.1 STEP .....   | 10 |
| 2.2.2 IFC .....  | 11 |
| 3 Stift-basierte Mensch-Maschine-Schnittstellen .....              | 11 |
| III Übersicht vorhandener Ansätze und Arbeiten.....                | 13 |
| 1 Allgemeine Skizzenwerkzeuge.....                                 | 14 |
| 1.1 Skizzieren in der Ebene.....                                   | 14 |
| 1.1.1 Flatland.....  | 14 |
| 1.2 Skizzieren im Raum.....  | 16 |
| 1.2.1 Teddy.....   | 16 |
| 1.2.2 Quick-Sketch.....  | 16 |
| 2 Architekturbezogene Skizzenwerkzeuge .....                       | 18 |
| 2.1 Architekturbezogenes Skizzieren in der Ebene.....              | 18 |
| 2.1.1 Electronic Cocktail Napkin.....                              | 18 |
| 2.2 Architekturbezogenes Skizzieren im Raum.....                   | 19 |
| 2.2.1 SketchPad+ .....   | 19 |
| 2.2.2 VR Sketchpad.....  | 20 |
| 2.2.3 Sketchhand+ .....  | 20 |
| 2.2.4 VRAM.....  | 21 |
| 2.2.5 DDDoolz.....   | 21 |
| IV Ein parametrischer Ansatz.....                                  | 22 |
| 1 Einleitung.....  | 22 |
| 2 Allgemeine Festlegungen.....                                     | 25 |
| 3 Vom Strich zur Wand.....   | 26 |
| 4 Von Wänden zu Räumen.....  | 28 |
| 5 Von Räumen zum Gebäude.....                                      | 29 |
| 6 Türen, Fenster und andere Bauteile.....                          | 30 |
| 7 Entwurfsunterstützende Zusatzinformationen.....                  | 33 |
| 7.1 Grundflächen.....  | 35 |
| 7.2 Wand- Fußboden- und Deckenoberflächen.....                     | 37 |
| 7.3 Fluchtwege und Brandabschnitte.....                            | 38 |
| 7.4 Ökologische Informationen.....                                 | 39 |
| 7.5 Räumliche Eindrücke.....                                       | 40 |
| 7.6 Zusammenhänge im technischen Ausbau.....                       | 41 |
| 7.7 Erschließung.....  | 41 |
| 7.8 Einbauten und Möblierung.....                                  | 41 |
| V Implementierung des Prototypen.....                              | 43 |

|                                       |    |
|---------------------------------------|----|
| 1 Einleitung.....                     | 43 |
| 2 Grundlegender Aufbau.....           | 44 |
| VI Diskussion und Ausblick.....       | 49 |
| 1 Zusammenfassung und Diskussion..... | 49 |
| 2 Ausblick.....                       | 50 |
| VII Anhang .....                      | 53 |
| 1 Literaturverzeichnis.....           | 53 |
| 2 Abbildungsverzeichnis.....          | 55 |
| 3 Danksagung.....                     | 56 |

## I Einleitung

Als wichtiges Hilfsmittel im planerischen Vorgehen des Architekten ist - spätestens seit der Prägung des Begriffes „graphidis scientia“ durch Vitruv - die zeichnerische Darstellung eines zu errichtenden Bauwerkes bekannt. Sie ist eine Vorwegnahme des Zukünftigen, Ausdrucks- und Kommunikationsmittel und physische Manifestierung mentaler Prozesse und Bilder. Sie dient dabei nicht nur der Kommunikation zwischen dem Nutzer oder Auftraggeber und dem Architekten. Sie ist auch mehr als eine Handlungsanleitung für den Ausführenden: Gerade in den frühen Phasen des Planungsprozesses ist die zeichnerische Darstellung vor allem Medium der Reflexion, Spiegel der Gedanken und Gefäß für Ideen. Unter den unterschiedlichen Strategien, Medien und Werkzeugen zur Entwicklung eines architektonischen Konzeptes nimmt die bildliche Darstellung eine zentrale Position ein.

Seit dem Aufkommen der elektronischen Datenverarbeitung gibt es zahlreiche theoretische Überlegungen, experimentelle Entwicklungen und industrielle Umsetzungen mit dem Ziel, die Arbeit des Architekten durch den Einsatz von Computern zu erleichtern, seine Ausdrucksmöglichkeiten zu erweitern und den Bauprozess als Ganzes zu rationalisieren. Mit der Etablierung grafischer Schnittstellen entstand zusätzlich zum ursprünglichen Zweck des Computers als Rechenmaschine zur Durchführung numerischer Operationen (etwa zur Berechnung bauphysikalischer oder statischer Eigenschaften eines Gebäudes) auch die Möglichkeit des Umgangs mit bildlicher Information. Gegenüber der analogen, physischen Bauzeichnung als wichtigstem Träger von Information im System der Architektur ist die Erzeugung bildlicher Darstellungen im System der Maschine jedoch nur eine mögliche, nachgeordnete Form der Repräsentation digitaler Information.

Dieser schwerwiegende Unterschied kann massive Auswirkungen auf die Arbeitsweise des Architekten haben. Der Gestaltung der Schnittstelle zwischen Mensch und Maschine muß daher als Vermittlerin zwischen den unterschiedlichen Systemen große Aufmerksamkeit und Sorgfalt gewidmet werden. Im Mittelpunkt der Betrachtung sollten bei jeder Bemühung zur Entwicklung solcher Schnittstellen der Mensch und die Qualität der von ihm geplanten Architektur stehen.

Ein Dolmetscher, der einen Satz unter möglichst umfassender Beibehaltung der Aussage in eine andere Sprache übersetzen möchte, benötigt dafür mehr als bloße Kenntnisse der syntaktischen und grammatischen Regeln beider Systeme: Die Interpretation des Gemeinten, des Sinns einer Aussage geht jeder eigentlichen Übersetzung voraus.

Da das Bezeichnete selbst (in diesem Fall die Architektur) im System der Maschine andere Bezeichnungen hat als in der analogen Skizze muß eine geeignete parallele Repräsentation gewählt werden.

In der vorliegenden Arbeit soll ein Ansatz vorgestellt werden, wie die Übersetzung einer Aussage aus dem System der Architektur in das System der elektronischen Datenverarbeitung vorgenommen werden könnte. Das Konzept stützt sich auf die Annahme, daß die Handskizze als eine Sprache der Architektur innerhalb gewisser Bereiche architektonischer Planung grundsätzlich für eine maschinelle Interpretation gut geeignet ist. Es wird der Prototyp einer Software vorgestellt, die als vermittelnde Instanz zwischen beiden Welten fungieren soll. Diese generiert in „Echtzeit“ architektonische Elemente aus den Eingaben einer Handskizze, die auf einem Tablet-PC erstellt wird. Die generierten Elemente werden dargestellt und auf Wunsch des Anwenders werden beispielhaft Zusatzinformationen errechnet und angezeigt. Die eigene Qualität der Handskizze soll dabei erhalten bleiben.

## 1 Gliederung der Arbeit

Die drei zentralen Gegenstände der Beschäftigung dieser Arbeit sind

1. die architektonische Skizze
2. die elektronische Datenverarbeitung in der Architektur
3. die Mensch-Maschine-Schnittstelle

Der Betrachtung ihrer jeweiligen Charakteristika und einer knappen Entstehungsgeschichte ist das erste Kapitel gewidmet.

Das zweite Kapitel stellt existierende Arbeiten aus dem Umfeld der Rechnerunterstützung in den frühen Phasen des architektonischen Entwurfs vor und diskutiert ihre Vor- und Nachteile

Im dritte Kapitel wird ein Verfahren zur parametrischen Echtzeitinterpretation von

architektonischen Handskizzen vorgestellt.

Das vierte Kapitel erläutert die prototypische Umsetzung des im dritten Kapitel vorgestellten Ansatzes.

Im fünften und letzten Kapitel schließlich erfolgt eine Diskussion der vorgestellten Arbeit und ein Ausblick auf mögliche zukünftige Entwicklungen.

## II Computergestütztes Skizzieren im architektonischen Kontext

### 1 Die Skizze im architektonischen Kontext

#### 1.1 Kurzer Geschichtlicher Abriß

Ein genauer Zeitpunkt der Entwicklung der zeichnerischen Darstellung als Mittel der architektonischen Planung läßt sich in der Architekturgeschichte nicht festmachen. Entstehungsgeschichtlich ist sie eng mit dem mit Entwicklung der Geometrie verbunden. Erst nach den wichtigen Entwicklungen auf diesem Gebiet, vor allem diejenigen Euklids, können zeichnerische Vorwegnahmen gebauter Umwelt angefertigt werden. Vitruv eröffnet mit seiner Definition der Architektur als Wissenschaft („architectura est scientia“) indirekt auch die systematische Auseinandersetzung mit der architektonischen Zeichnung, der „graphidis scientia“. Wichtige theoretische Betrachtungen zur Rolle der Zeichnung in der Architektur finden sich bei Alberti ([ALBERTI12]): „Die ganze Baukunst setzt sich aus 'lineamenta' und 'structura' zusammen.“. Lineamenta ist dabei das Liniengefüge in der Architekturdarstellung. „Durch die Verbindung und das Zusammenfügen von Linien und Winkeln würde die Erscheinung eines Bauwerks erfasst" und "die ganze Form und Figur eines Gebäudes [würde] schon in den Zeichnungen selbst festgelegt“ (zitiert nach [OECHSLIN02]) Spätestens jedoch seit Andrea Palladios „Vier Bücher über die Architektur“ [PALLADIO93] sind architektonische Zeichnungen fester Bestandteil der Mitteilung über Architektur. Im Zusammenhang mit der Entwurfsskizze sind die zeichnerischen Methoden interessant, die bei der Vorstellung rationalistischer und industrieller Entwurfsmethoden durch Jean N.L. Durand in Erscheinung treten. Jean Cousin setzt das Mittel der Netzwerkdarstellung, wie sie aus der Graphentheorie der diskreten Mathematik bekannt sind, zur Darstellung topologisch-räumlicher Zusammenhänge innerhalb von Grundrissen ein.

Zum Wandel der Bedeutung der architektonischen Zeichnung unter dem zunehmenden Einfluss von C(A)AD finden sich ausführliche Betrachtungen u.a. in [DONATH88], [MITCHELL94], [SCHMITT93] und [SCHMITT96]

### 2 Modelle zur computergestützten Beschreibung von Architektur

In der Literatur finden sich eine große Anzahl von Modellen zur Beschreibung



bauwerksrelevanter Daten und deren mögliche Abbildung und Verwaltung mithilfe der Informationstechnologie. Da die unterschiedlichen Modelle sich stark in ihren Ansätzen, Einsatzbereichen und Detaillierungsgraden unterscheiden, sollen ausgewählte Vertreter nach einer knappen Übersicht in Hinblick auf ihre Bedeutung in den frühen Phasen des architektonischen Entwurfs untersucht werden.

## 2.1 Der Begriff des Modells in der Bauwerksbeschreibung

Der Begriff des Modells, dessen Ursprung im lateinischen „modulus“ zur Bezeichnung des Grundmaßes aller Teile eines Tempels liegt ([DUDEN89]) wird zur Abstraktion und verallgemeinernden Reduktion zur Beschreibung von „Wirklichkeit“ herangezogen. (Ausführlich in [STACHOWIAK73] ) Im Zusammenhang mit der Beschreibung von Bauwerken<sup>1</sup> finden sich dazu bei [PETZOLD01], der sich auf [LIEBICH93] bezieht, folgende Aufteilung in grobe Kategorien von Modellen:

- „Beschreibende Modelle“ zu rein textlichen Umschreibungen
- „Analoge Modelle“ als Analogien zu Metainformationen z.B. funktionaler Art in Form von Diagrammen, und Schemata
- „Symbolische Modelle“ zur Beschreibung bauwerkspezifischer Eigenschaften bspw. bauphysikalischer Natur mithilfe von Formeln und Algorithmen
- „Ikonische Modelle“ zur Charakterisierung von räumlichen Grundstrukturen unter Vernachlässigung physikalischer Größen
- „Wirklichkeitsnah-fotografische Modelle“ als Abbild vorhandener Substanz
- „Physisch-nachbildende Modelle“ als maßstabsgerechte Repräsentation in Architekturmodellen
- „Rechnerintern-nachbildende Modelle“ zur hybriden Beschreibung in ikonischer und nachbildender Form innerhalb des Rechners.

Petzold identifiziert weiterhin vier Kategorien von Information, die als Datenstrukturen in einem objektorientierten Bauwerksmodell auftauchen können.

- geometrische
- formalisierbare
- informale

---

<sup>1</sup> Petzold bezieht sich hier speziell auf Modelle für die Beschreibung von Bauwerken in der Bauaufnahme, während Liebich die Entwurfsphase untersucht.

und

– relationale

Informationen.

Eine umfangreiche Klassifizierung der Eigenschaften von Bauwerken findet sich in [LEEUEWEN99]: Van Leeuwen schlägt hier ein Konzept zur Beschreibung von Bauwerken vor, das seine historischen Wurzeln in Produktmodell-Ansätzen des Maschinenbaus hat. Das zentrale Konzept des sog. „Feature Based Modeling“ (FBM) in der Architektur ist dabei die Entkoppelung von Eigenschaftssätzen aus den Hierarchiestrukturen der klassischen Objektorientierung. Er definiert:

*A Feature is an autonomous, coherent collection of information, with semantic meaning to a designer and possibly emerging during design, that is defined to formalise a design concept at any level of abstraction, either physical or non-physical, as part of a building model.*

Eine Inventarisierung und Klassifizierung der „Features“ in der Architektur führen zu dem hier dargestellten Schema:

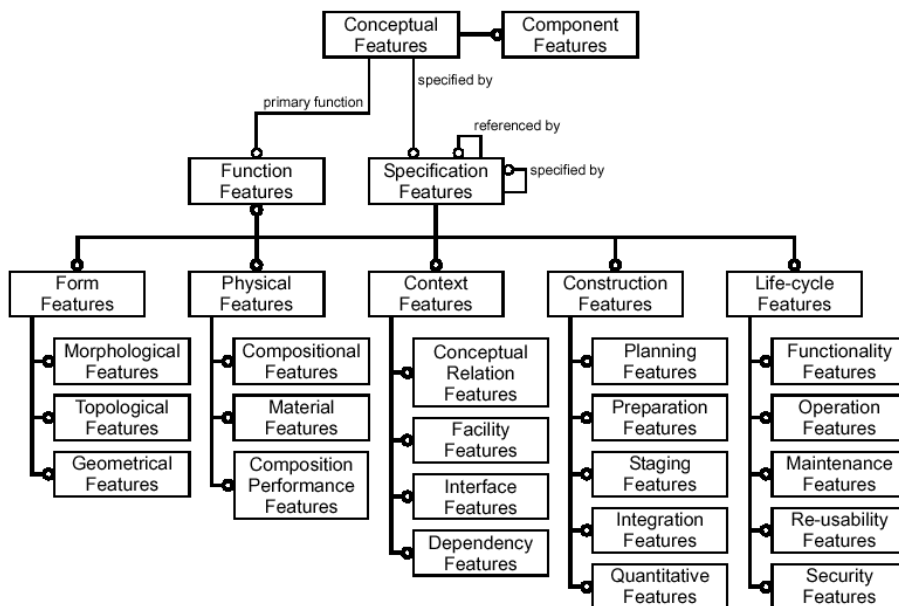


Abb 1 architektonische „Features“ nach Leeuwen.

## 2.2 Etablierte Standards zur Beschreibung von Gebäudemodellen

### 2.2.1 STEP

Die Bemühungen, ein herstellerunabhängiges Datenformat für den Austausch von Produkt-Daten im Bereich des Computer Aided Design zu etablieren begannen 1984 mit der Entwicklung des STEP-Formates (Standard for Exchange of Product model

data). 1994 erfolgte die offizielle Zertifizierung, die im ISO 10303 festgehalten ist. Das Austauschformat ist stark bauteilorientiert, läßt aber auch andere Formen der Datenorganisation zu. Neben geometrische Beschreibungen, die sich auf BRep-Beschreibungen gründen, können beliebige Attribute an Bauteile angehängt werden. Als wichtige Modellierungssprache spielt beim Umgang mit STEP-Daten EXPRESS eine zentrale Rolle, die neben der Modellierung von Attributen auch relationale und regelbasierte Verbindungen einzelner Bauteile oder deren Konzepte zuläßt.

### 2.2.2 IFC

Ein grosses Problem bei der Beschreibung von Bauwerksdaten durch das STEP-Format ist seine große Flexibilität. Da es ebenso für die Beschreibung von Produktmodellen aus dem Maschinenbau sowie dem Fahrzeug – und Flugzeugbau und anderen geeignet ist, entstand 1995 die IAI (International Alliance for Interoperability). Als ein loser Zusammenschluß von Weltweit mehr als 600 Unternehmen und akademischen Einrichtungen, aus der Baubranche kümmert sich diese Konsortium um eine Standardisierung von Bauwerksbeschreibungen in den sog. IFC (Industry Foundation Classes). Verschiedene Versionen des Standards (IFC 1.0 – IFC 3.0), wurden in zahlreiche Softwareprodukte integriert. Unter ihnen finden sich die wichtigsten Hersteller von C(A)AD- Produkten, wie AutoDesk, Nemetschek, Graphisoft und Bentley. Als ein wichtiges technisches Format rückt zunehmend die Verwendung von XML in den Mittelpunkt.

## 3 Stift-basierte Mensch-Maschine-Schnittstellen

Den Grundstein für die direkte Interaktion zwischen Mensch und Maschine vermittelt eines stiftähnlichen Eingabegerätes legte Ivan Sutherland 1963 in seiner Doktorarbeit „Sketchpad“ [SUTHERLAND63]. In der Arbeit wurden zahlreiche grundlegende Schnittstellen-Metaphern wie das „Anfassen und Bewegen“, Vergrößern von Objekten und andere Möglichkeiten der direkten Objektmanipulation durch einen Lichtstift vorgestellt. „It contained the seeds of myriad important interface ideas“ [MYERS98]. Spätere Entwicklungen, wie die Einführung der Maus durch Doug Engelbart und andere im Xerox Parc machten die unmittelbare Manipulation grafischer Objekte für den Massenmarkt tauglich und setzten bis heute gültige Standards in der Mensch-

Maschine-Interaktion. Die ersten Zeichenprogramme, die die Maus oder das ursprünglich im Auftrag der ARPA entwickelte Grafik-Tablett als Eingabegeräte verwendeten, entstanden Mitte der siebziger Jahre. Stifte für die Interaktion im Raum, wie der „Lincoln Wand“ (Ermittlung der räumlichen Position durch Ultraschall) von Larry Roberts wurden für den Einsatz im militärischen Konstruktionsbereich und das Modellieren von molekularen Strukturen bereits in den sechziger Jahren entwickelt. Ausführliche Betrachtungen zur Geschichte der Mensch-Maschine-Schnittstellen finden sich in [RUDISILL95].

### III Übersicht vorhandener Ansätze und Arbeiten

Das Bestreben, skizzenbasierte computergestützte Methoden zur Unterstützung den frühen Phasen des architektonischen Entwurfes einzusetzen, hat in der Geschichte des Computer Aided Architectural Design eine lange Tradition. Während es zahlreiche theoretische Überlegungen gibt, wie sich den dabei auftretenden Problemen zu nähern sei, sind die (prototypischen) Umsetzungen eher rar. Eine der größten Schwierigkeiten, die sich dabei stellt ist das Übertragen abstrakter, konzeptioneller Ideen in „nackte“ Zahlen und Fakten. Auf der einen Seite ist der entwerfende Architekt bestrebt einen „unscharfen“ Spielraum in seinen ungefähren Festlegungen auf dem Papier offen zuhalten, auf der anderen Seite sind gerade diese Unschärfen hinderlich bei der Ermittlung konkreter Konsequenzen dieser Festlegungen durch den Computer.

Vor die eingehende Betrachtung von computergestützten Skizzierwerkzeugen – seien es nun allgemein nutzbare oder auf den architektonischen Kontext angepasste – sei an dieser Stelle eine grundsätzliche Kategorisierung dieser Hilfsmittel vorgenommen:

Skizzierwerkzeuge sind auf der einen Seite grob in

a) ebene

b) räumliche ,

auf der anderen Seite in

a) diskrete und symbolische

b) parametrische

unterteilbar. Die beiden Unterscheidungen können in beliebigen Kombinationen auftreten (Also gibt es sowohl ebene symbolische als auch räumliche symbolische). In den nachfolgenden Abschnitten werden einige prominente Vertreter dieser unterschiedlichen Kategorien untersucht.

## 1 Allgemeine Skizzenwerkzeuge

### 1.1 Skizzieren in der Ebene

#### 1.1.1 Flatland

Flatland [MYNATT99] ist ein Gemeinschaftsprojekt der Gorgia Institute of Technology, der University of Tokyo und dem Xerox Palo Alto Research Center.



Abb. 1 Das Whiteboard von Flatland

Kernstück der Entwicklung ist ein analoges Whiteboard, das im Hintergrund die auf ihm angefertigten Zeichnungen interpretiert. Als Ein- und Ausgabegerät wird eine sensitive Wandtafel, eine Stift mit Knopf und ein Projektor verwendet.

Die Interaktion ist in drei Haupthandlungen aufgeteilt:

1. Freihandeingabe von Strichen (diese dürfen sich nicht überschneiden)
2. „Metastriche“ bei gedrückt gehaltenem Stiftknopf zur Manipulation der Strichskizzen (wie bsp. Translation)
3. Ein Pie-Menü, das mit einem einfachen „Metapunkt“ aufgerufen wird und Funktionen wie vor- und rückgängig machen, speichern und löschen, vor allem aber die Wahl der jeweiligen Interpretationsdomäne („Verhaltensweise“) ermöglicht.

Verschiedene „Verhaltensweisen“ übernehmen die eigentliche Interpretation des Eingebenen. Beispielhaft implementiert sind die Ausführung einfacher Rechnungen, sowie die Anfertigung von Anfahrtsskizzen, Aufgabenlisten und unspezifischen Zeichnungen. Die jeweilige Interpretationsdomäne wird durch einen „Assistenten“ oder eine „Muse“ in form eines kleinen Comic-Tieres in der rechten oberen Ecke der jeweiligen Bounding Box einer Zeichnung angezeigt. (Ähnlich der zwei Jahre später in die Office-Linie der Microsoft Produkte eingeführten „Büroklammer“ *Clippy*)



Abb. 2 Vier Beispiel-Verhaltensweisen in Flatland

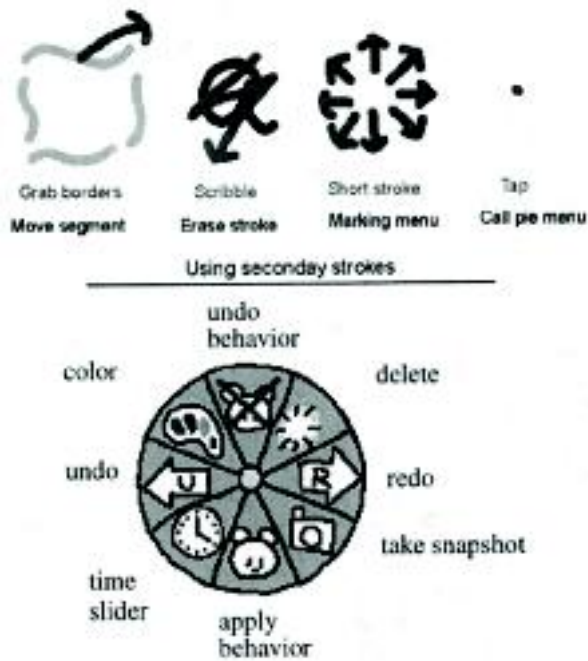


Abb. 3 Interface- Metaphern in Flatland

Auf eine Handschrifteninterpretation wurde – abgesehen vom grafischen Rechner – aus Gründen der Fehleranfälligkeit und der dadurch erschwerten Benutzbarkeit verzichtet.

Das eigentliche Skizziermodul basiert auf der vorangegangenen Arbeit „Pegasus“ [IGARASHI98] des am Projekt beteiligten Takeo Igarashi. Hierbei handelt es sich um ein „voraus interpretierendes“ Zeichenwerkzeug, das dem Zeichner ähnlich den

Fangfunktionen in Autodesk's AutoCAD während des Zeichnens zu anderen Objekten gefundene Beziehungen anzeigt (Rechtwinkligkeit, Parallelität, Eckenfang usw.) und Striche der Auswahl entsprechend anpasst.

Beachtenswert an dieser Arbeit ist das Bemühen um eine möglichst natürliche Interaktion. Die offensichtliche Notwendigkeit, die jedoch die Autoren dazu veranlassten den Nutzer zwischen verschiedenen Interpretationskontexten manuell umschalten zu lassen, deuten darauf hin, daß selbst bei diesen scheinbar einfachen Aufgabenstellungen die Notwendigkeit der Ein- und Abgrenzung für brauchbare Interpretationsergebnisse nötig ist.

## 1.2 Skizzieren im Raum

### 1.2.1 Teddy

Die Applikation „Teddy“ von Igarashi et al [IGARASHI01] ist unter den bildschirmbasierten Skizzierwerkzeugen der letzten Jahre eine Standardreferenz geworden. Grund dafür ist unter anderem die einfache Handhabung: Mit der Maus wird zunächst ein geschlossenes Grundprofil gezeichnet das als Silhouette eines Rotationskörpers fungiert, der als geometrischer Ausgangspunkt für die weitere Zeichnung dient. Durch einfache Gesten können an den Ausgangskörper weitere Blobs angefügt werden, Teile herausgeschnitten oder Körper durchtrennt werden. Die Geometrien können jederzeit von allen Seiten in einem Sketch-Renderer betrachtet werden. Sämtliche Interaktionen basieren auf der jeweiligen Silhouette der aktuellen Nutzerperspektive. Da sämtliche entstehenden Geometrien aus Rotationskörpern zusammengesetzt sind eignet sich Anwendung nur im begrenzten Maß für den Einsatz im architektonischen Kontext.

### 1.2.2 Quick-Sketch

„Quick-Sketch“ ist die Implementierung der langjährigen Forschungsarbeiten von Lynn Eggli, Ching-yao Hsu, Beat Brüderlin und Gershon Elber [EGGLI97]. Obwohl es vornehmlich auf den Maschinenbau ausgerichtet ist eignet es sich prinzipiell auch für die Anwendung im architektonischen Kontext. Ausgangspunkt der Erstellung dreidimensionaler Objekte ist die Anfertigung einer groben 2D-Skizze. Die eingehenden Striche werden anhand vordefinierter Constraints (Rechtwinkligkeit,



Parallelität, Kreisbogen usw.) „gesäubert“, zueinander in Beziehung gesetzt und in entsprechende Geometrien umgewandelt. Die Striche - die sich hier ebenfalls nicht überschneiden dürfen - werden an ihren Eckpunkten zusammengesetzt, und geschlossene Konturen erkannt.



Abb. 4 Quick-Sketch: "Zittrige" Eingangsskizzen

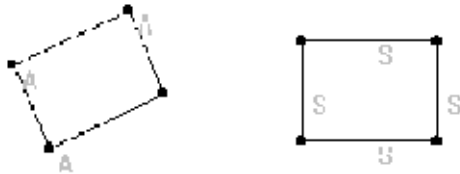


Abb. 5 Quick-Sketch Bereinigte Eingangsskizzen

Um aus der so entstandenen Grundskizze ein dreidimensionales Objekt zu erstellen stehen zwei verschiedene Möglichkeiten zur Wahl: Sweeping, das Extrudieren eines geschlossenen Profiles entlang einer Achse und Ruled Surfaces, Oberflächen die zwischen zwei begrenzenden Kurven entstehen. Zusätzliche Modifikationen an dem so entstandenen Körper lassen sich durch projizierte Striche vornehmen: Um ein Loch zu erstellen wird dieses perspektivisch passend auf das Werkstück gezeichnet, das System errechnet aus der momentan für den Nutzer sichtbaren Perspektive und der Strichkontur die Lage im lokalen Koordinatensystem des Objektes und „schneidet“ mittels bool'scher Operation ein entsprechend der Tiefe der Ausgangsgeometrie extrudiertes Objekt aus der Geometrie.

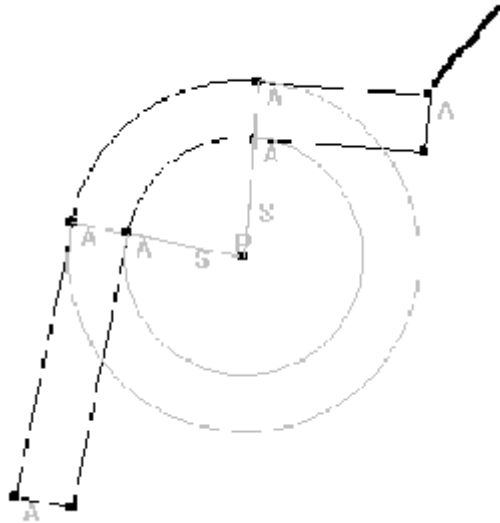


Abb. 6 Quick-Sketch: "gesäuberte" Eingangsskizze mit Darstellung der verwendeten Constraint. Rechts oben die eingezeichnete Extrusionsachse

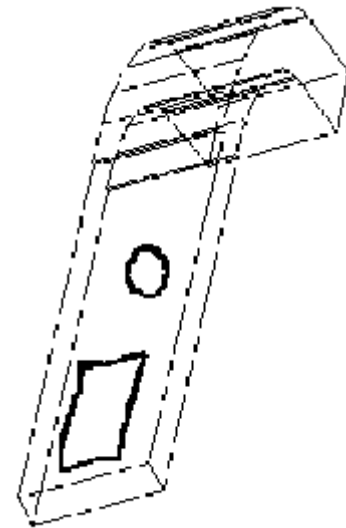


Abb. 7 Quick-Sketch: perspektivische eingezeichnete Features

## 2 Architekturbezogene Skizzenwerkzeuge

### 2.1 Architekturbezogenes Skizzieren in der Ebene

#### 2.1.1 Electronic Cocktail Napkin

Die „elektronische Cocktail Serviette“ [GROSS96] von Mark D. Gross und Ellen Yi-Luen Do aus dem Jahr 1996 ist eine der ersten wichtigen Forschungsarbeiten zum Thema computergestütztes Skizzieren im architektonischen Kontext. Hardwareseitig auf WACOM-Tablets oder Apples Newton-Serie gestützt setzt es auf Mustererkennung zur Interpretation der Nutzereingaben. Für eine Sequenz von Strichen wird dabei das bildschirmorthogonale minimal umgebenden Rechteck (*minimal bounding rectangle*) ermittelt und in ein 3x3 Raster zerlegt. Aus der Reihenfolge, in der sich der Stift durch die einzelnen Rasterfelder bewegt entsteht eine Ablaufsequenz, die mit einem symbolischen Datum assoziiert werden kann<sup>2</sup>. Durch die relationalen Kombination einzelner Symboltypen („waagerechter Strich über Kreis“ etc.) kann eine Vielzahl verschiedener Gesten und Symbole voneinander abgegrenzt werden.

2 Die auf Teitelmans frühe Arbeiten aus dem Jahr 1964 zurückgehende Buchstabenerkennung wird teilweise noch heute in Handheld-Geräten und verschiedenen OpenSource Projekten zur Schrifterkennung eingesetzt. Die von Gross und Do verwendete Methode ist um die Eingabe mehrerer Striche (*multiple strokes*) erweitert

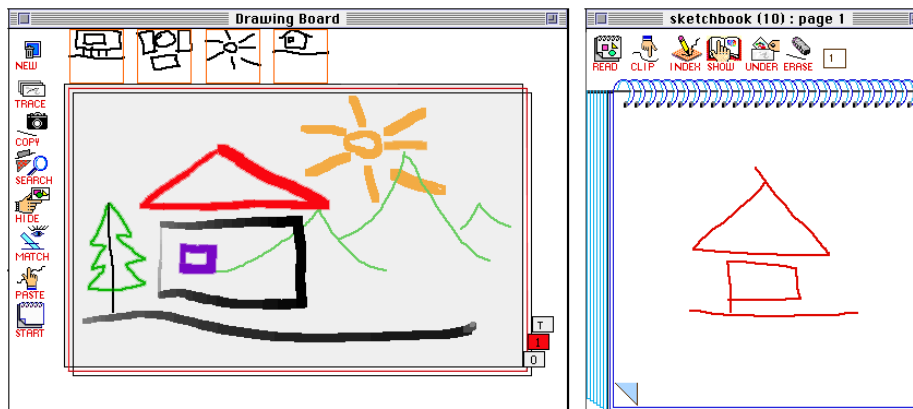


Abb. 8 Cocktail Napkin: Das Interface des Prototypen

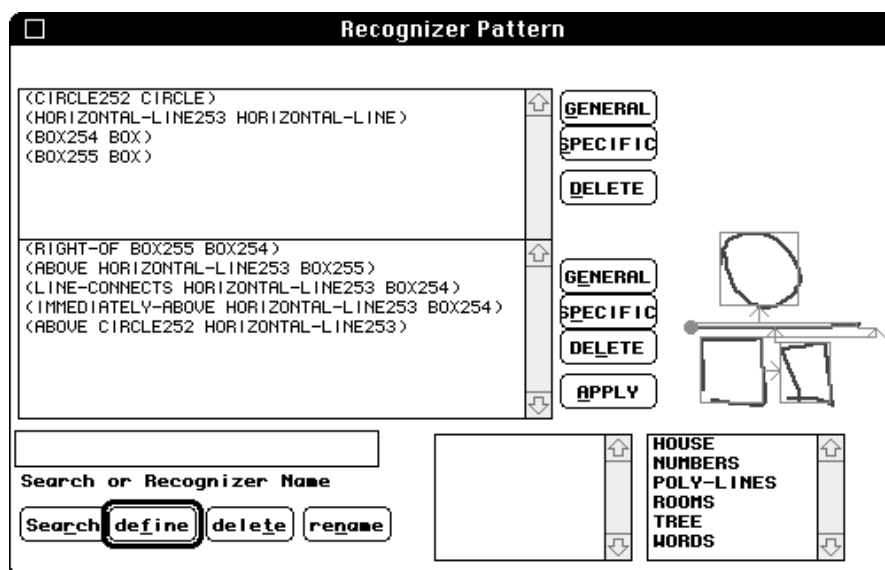


Abb. 9 Cocktail Napkin: Konsole für die Symbolsprache zum Erstellen und vergleichen von Gesten

## 2.2 Architekturbezogenes Skizzieren im Raum

### 2.2.1 SketchPad+

Die 1998 von M. Piccolotto [PICCOLETT098] im Rahmen einer Masterarbeit entwickelte Anwendung „SketchPad+“ ermöglicht dem Nutzer dreidimensionale Architekturskizzen auf einem großen Tablett anzufertigen. In einem Framework, das auf einem Zusammenspiel von Inventor und dem ACIS-Solid-Modeling-Kernel basiert, zeichnet der Nutzer zunächst in der Ebene erste B-Splines und NURBS-

Kurven. Durch Kippen der virtuellen Arbeitsebene können aus den erzeugten Kurven in einem nächsten Schritt mittels Extrusion Flächen generiert werden. Das Erzeugen von Körpern sowie deren Manipulation bspw. durch die Anwendung bool'scher Operationen sind in weiteren Teilen des Systems möglich. Durch die Anbindung eines externen Radiosity Render-Pakets können aus den erzeugten Szenerien Global Illumination Lösungen erschaffen werden.

### 2.2.2 VR Sketchpad

Das von Ellen Yi-Luen Do entwickelte VR Sketchpad [DO01] führt die in „Electronic Cocktail Napkin“ begonnenen Ansätze in die dritte Dimension fort. Hier wie dort besteht die Anwendung im Kern aus einer Symbolerkennung durch ein diskretes 3x3 Raster. Erkannte Symbole werden in eine dreidimensionale Umgebung aus der Symbolbibliothek der kommerziellen Anwendung Form-Z eingefügt. Die so zusammengefügte Szenerie kann als VRML-Welt exportiert und in einem herkömmlichen 3D-Browser betrachtet werden. Nichterkannte Strichsequenzen werden senkrecht mittels von Standardwerten extrudiert und ebenfalls als dreidimensionale VRML-Geometrien exportiert. Eine weiterführende Architekturfunktionalität ist nicht implementiert.

### 2.2.3 Sketchhand+

Die als Diplomarbeit an der Bauhaus-Universität entstandene Anwendung „Sketchhand+“ von Hartmut Seichter [SEICHTER02] läßt den Nutzer mithilfe eines frei im Raum bewegbaren Grafiktablets die Umrisszeichnungen architektonischer Objekte zeichnen. Durch eine an einer Datenbrille befestigte Kamera wird die natürliche Umgebung des Zeichnenden – z.B. ein Städtebaumodell – dem Nutzer als Videobild dargestellt. An Stellen, an denen durch Bilderkennungsalgorithmen bestimmte Marker in Form bedruckter Pappschilder erkannt werden<sup>3</sup>, werden die Videobilder der Kamera mit den erstellten Geometrien überlagert. Hauptaugenmerk der Entwicklung lag jedoch auf der Entwicklung neuer Kollaborationstechniken, mithilfe derer entfernt voneinander am Entwurf Beteiligte über diesen kommunizieren können.

---

<sup>3</sup> diese dienen zur räumlichen Lagebestimmung, die Erkennung ist Teil der verwendeten freien API „AR Toolkit“

#### 2.2.4 VRAM

Das an der Bauhaus-Universität Weimar entwickelte Projekt „VRAM – Virtual Reality Aided Modeller“ [REGENBRECHT00] basiert auf unmittelbarer Eins-zu-eins-Manipulation und -Erstellung von Objekten. Neben zahlreichen Hilfsmitteln zum Betrachten und Navigieren von virtuellen Welten werden dem Nutzer Werkzeuge zur Entwicklung neuer Geometrien innerhalb der virtuellen Umgebung angeboten. Unter anderem stehen Standard-Editier-Funktionen wie Rotieren, Skalieren, Verschieben, die Extrusion von Polygonen und das Einfügen von Objekten aus externen Bibliotheken zur Verfügung. Diese können aus entsprechenden Listenmenüs, die dem Nutzer auf unterschiedliche Art in die virtuelle Umgebung eingeblendet werden oder durch eine implementierte Gestenerkennung im dreidimensionalen Raum eingefügt werden.

#### 2.2.5 DDDoolz

Die an der TU Eindhoven von Bauke de Vries et al entwickelte Applikation „DDDoolz“ [VRIES00] ermöglicht durch Extrusion in diskreten Schritten kubische Geometrien zu erzeugen. Entstehende Voxel können als Punkte, Kanten oder Flächen mit einem Stift „gegriffen“ und in eine der sechs orthogonalen Richtungen des jeweiligen Objektes gezogen werden. Wie in anderen zuvor vorgestellten Werkzeugen dient der Prototyp eher der Formfindung als der frühzeitigen Beurteilung anderer Parameter, die im architektonischen Entwurf eine Rolle spielen.

## IV Ein parametrischer Ansatz

### 1 Einleitung

Wie im Ausgang des vorhergehenden Kapitels erörtert, bergen viele der im architektonischen Kontext vorhanden Ansätze unterschiedliche Probleme und Schwächen in sich. Den dort besprochenen prototypischen Lösungsansätzen ist gemein, daß sie durchweg eher der formalen Gestalt als den funktionellen Randbedingungen der architektonischen Praxis verpflichtet sind. So hilfreich und nützlich ein Werkzeug zum Entwurf von Freiformflächen als raumbegrenzende architektonische Elemente in den frühen Phasen des Entwurfes bei vielen Gelegenheiten auch sein mögen: der Regelfall in der täglichen Praxis des Bauens sind sie kaum. Hier, unter zusehends schwerer werdenden Bedingungen und einer immensen Zahl vom Planenden zu berücksichtigender Parameter und Anforderungen sind es oft die einfachen Dinge, die eine große Erleichterung darstellen könnten.

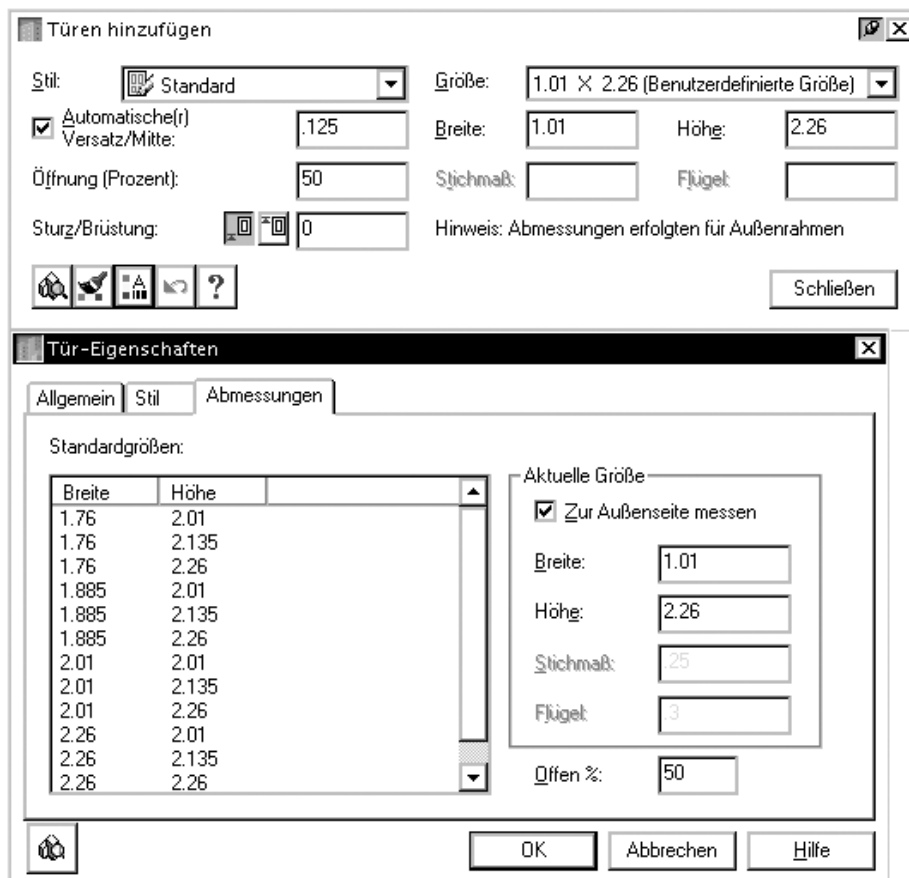


Abb. 10 Konfigurationsdialoge für die Erstellung einer Tür im AutoCad Architectural Desktop

Ein massives Problem beim Einsatz computergestützter Techniken in den Frühen

Phasen des Entwerfens stellt die Integration der Mensch-Maschine-Schnittstelle in den Arbeitsprozess des Architekten dar. In Kommunikationsprozeß zwischen „ich“ und „mir“ (Orskov) stellen sich bei der Benutzung von Softwareprogrammen gegenüber dem Skizzenpapier nicht nur die physikalischen Umwege der meisten Hardwarekonfigurationen (indirekter Weg Maus-> Projektion an einem anderen Ort in einem anderen Maßstab). Dramatischer noch sind die zahlreichen Menüs, Knöpfe, Tastenkombinationen und andere Interface-Metaphern, die ausschließlich im Zeichenvorrat der Maschine, nicht jedoch in dem der architektonischen Skizze vorkommen. So muß sich der Zeichnende an ein kompliziertes Geflecht von Regeln, Kommunikationsprotokollen und Benutzungsvorschriften halten, die zusätzliche kognitive Last bedeuten [HUTCHINS86].

Anstatt monotoner, langwieriger Aufgaben wie etwa der Ermittlung von Raumgrößen entledigt zu werden, fallen neben der eigentlichen Aufgabe – der Suche nach einer architektonischen Lösung – zusätzliche Problematiken an. Der hier vorgestellte Ansatz verfolgt deshalb die Strategie möglichst keine andere Schnittstelle zwischen das „Ich“ und das „Mir“ zu stellen, als eine Umsetzung von Stift und Zeichenpapier im Computer. Hardwareseitig am nächsten kommt diesem Vorhaben der Tablet-PC: Bestehend aus einem Stift und einem Untergrund der Gezeichnetes unmittelbar am Ort des Entstehens darstellen kann ist diese Konfiguration noch besser geeignet als Kombinationen aus Grafik-Tablets und herkömmlichen Bildschirmen.



Abb. 11 Tablet-PC (Compaq TC 1000)

Das Interface der Softwareschnittstelle – zentraler Gegenstand dieser Arbeit – sollte soweit wie nur irgend möglich in den Hintergrund treten: Dort, wo bei herkömmlichen CAD-Werkzeugen fest vorgeschriebene Folgen von Abläufen dominieren, soll hier die freie Zeichnung im Vordergrund stehen. Eine „natürliche“ Reaktion des Zeichenmediums auf die Handlungen seines Nutzers hat oberste Priorität, alle Rückschlüsse, die die Software aus der Eingangszeichnung über ihre Bedeutung im architektonischen Kontext zieht, sollten „unauffällig“ im Hintergrund erfolgen. Der Wert der physikalischen, analogen Skizze soll damit erhalten bleiben und der Computer seiner Rolle als *mögliches zusätzliches* (und eben nicht dominierendes) Werkzeug gerecht werden.

Zahlreiche Wesensmerkmale der physikalischen Handskizze stehen dabei allein der Abbildung im diskreten System des Rechners diametral entgegen. Zu den vier wichtigsten zählen:

1. Die hohe Auflösung der Eingabegeräte (dies betrifft sowohl den „unendlich fein“ aufgelösten ebenen Raum des Papiers als auch zusätzliche Qualitäten wie Andruck, Abrieb und Anstellwinkel des verwendeten Zeichengerätes)
2. Die zeitliche Unmittelbarkeit der Auswirkung der Handlung des Zeichnenden
3. Das Licht- und Farbmodell sowie weitere optische Eigenschaften des



Ausgabemediums (passive Beleuchtung mit subtraktivem Farbmodell des Papiers gegenüber aktiver Lichtquelle mit additivem Farbmodell im RGB-Farbraum des Computerbildschirms u.a. siehe auch [GOLDSTEIN96])

4. Die multi-sensorischen Eigenschaften des Mediums (vornehmlich haptische und akkustische, aber auch olfaktorische und thermale Eigenschaften [FERGUSON77])

Weitaus schwerer als diese Problematiken wiegen jedoch die die Problemfelder im Zusammenhang mit der Interpretation des Gezeichneten im architektonischen Kontext. Die fundamentalen Schwierigkeiten hierbei sind

1. Großer Umfang in Syntax und Semantik
2. Unschärfe als Prinzip
3. Mehrfachkodierung
4. Große Bandbreite individueller Stile und Symboliken
5. Intersubjektive Unterschiede in der Deutung

Während viele Problemstellungen auf dem Gebiet der Reproduktion der Qualitäten von Skizzen im Rechner oft nur hardwareseitig angegangen werden können (und damit aus dem Rahmen dieser Arbeit fallen) können für die zweite Gruppe von Problemen einige hilfreiche Vereinfachungen vorgenommen werden.

## 2 Allgemeine Festlegungen

Um mit einer Skizze zu beginnen sollten dem Anwender – im Gegensatz zu vielen existierenden Anwendungen – möglichst wenig Schritte zum „Einrichten“ der Zeichnung abverlangt werden, da dies oft schon einen großen Hinderungsgrund zum Einsatz des Rechners in den frühen Phasen des Entwurfs darstellt<sup>4</sup>. Die Möglichkeit „einfach drauf los“ zu zeichnen sollte oberste Priorität haben. Das hier vorgeschlagene System sollte die Möglichkeit haben, nachträglich die Maßstäblichkeit und die Standardeigenschaften von Bauteilen, wie Geschosshöhen, Wandaufbauten und Türbreiten zu ändern. Dadurch kann eine einfach Strichzeichnung nachträglich ein

---

4 Wie an vielen anderen Punkten auch, ist hier die geforderte Präzision und Detailierung, die dem Planer abverlangt werden das Hauptproblem. Diese ist zwar zu späteren Zeitpunkten, wie der Anfertigung eines Werkplanes zwingend erforderlich, in den bewusst ungenau gehaltenen anfänglichen Stadien jedoch hinderlich. Versuche In Anwendungen beide Aspekte (den ungenauen und den genauen) in einer einzigen Oberfläche unterzubringen, wie etwa im ADT von Autodesk sind oft eher unglücklich, da Präzisionsmethoden nachträglich ein „unscharfes“ Profil aufgezwungen wird

verschiedene Kontexte gebracht werden.

### 3 Vom Strich zur Wand

Um von einer Menge diskreter Punkte, wie sie von einem Grafiktablett oder einem Tablett-PC empfangen werden können, eine „gemeinte“ Linie und damit für diesen Kontext die Lage von einer Wand oder Wandzügen zu ermitteln, eignet sich, wie Jenkins et al gezeigt haben [JENKINS92] die Diskussion von approximierten B-Spline Kurven. (Dieser Ansatz wurde ebenfalls Grundlage für das in III besprochene „Quick-Sketch“ von Egli et al [EGGLI97] gewählt)

Dabei wird zunächst aus einer Menge von Punkten durch Approximation eine genährte B-Spline Kurve errechnet (siehe dazu [BOOR78]). Durch unterschiedliche Wichtung der Knotenvektoren können dabei „zittrige“ Striche innerhalb wählbarer Toleranzen geglättet werden.

Mehrfach übereinander gezeichnete Linien sind – in Abhängigkeit des persönlichen Zeichenstils – in architektonischen Skizzen häufig anzutreffen. Sie werden im Prozess der Kommunikation zwischen dem „Ich“ und dem „Mir“ auf dem Papier häufig zur schrittweisen Annäherung an eine Lösung verwendet. Der menschliche Wahrnehmungsapparat reduziert dabei solcherlei Ansammlungen von Strichen automatisch auf eine Ideallinie. Um diesen Effekt mithilfe von Rechenverfahren nachzubilden, kann ein Strichkonglomerat folgendermaßen ausgewertet werden: Aus allen Strichen, die sich in einem gewissen Abstand zueinander befinden, und die außerdem innerhalb gewisser Toleranzen parallel zueinander sind, wird dabei das mathematische Mittel gebildet<sup>5</sup>. Für eine weitere Verfeinerung des Verfahrens können zusätzlich der Andruck der jeweiligen Striche (Schwärzungsgrad) und die zeitliche Erstellungsreihenfolge in Betracht gezogen werden: Dabei fließen diejenigen Striche (Punkte in einer Punktwolke) durch entsprechende Multiplikatoren stärker in die Mittelwertsberechnung ein, die entweder durch besonders festes Aufdrücken betont worden sind oder zu einem späteren Zeitpunkt korrigierend über die vorhanden Striche gezeichnet werden. Zu überprüfen ist, ob dabei das tatsächliche mathematische Mittel als Resultierende gewählt werden sollte (invasive Methode),

---

5 Dabei können entweder die einzelnen Striche herangezogen werden oder die einzelnen Punktwolken, aus denen sie zunächst konstruiert wurden.

oder ob diejenige vorhandene Strichsequenz zu weiteren Auswertung herangezogen werden sollte, die diesem mathematischen Mittel am nächsten ist (minimalinvasive Methode)<sup>6</sup>

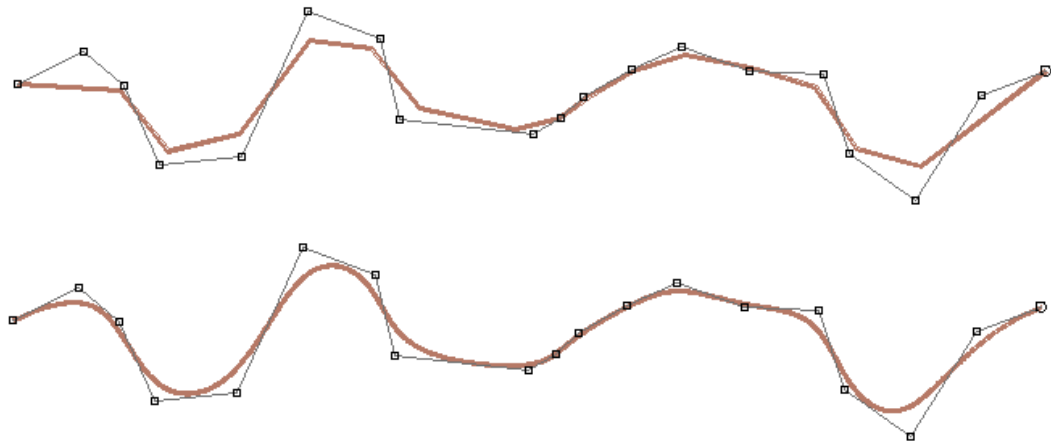


Abb. 12 Ungeglättete und geglättete approximierte Spline-Kurve mit Gewichten

Durch Ermittlung und Analyse der Extrema und Nullstellen der ersten und zweiten Ableitungen (Hodographen) können in einem zweiten Schritt dann Änderungen in Krümmung und Richtung ermittelt werden. Aus diesen Daten lassen sich nun wiederum Rückschlüsse auf „gemeinte“ Ecken innerhalb eines einzelnen Kurvenzugs ziehen.

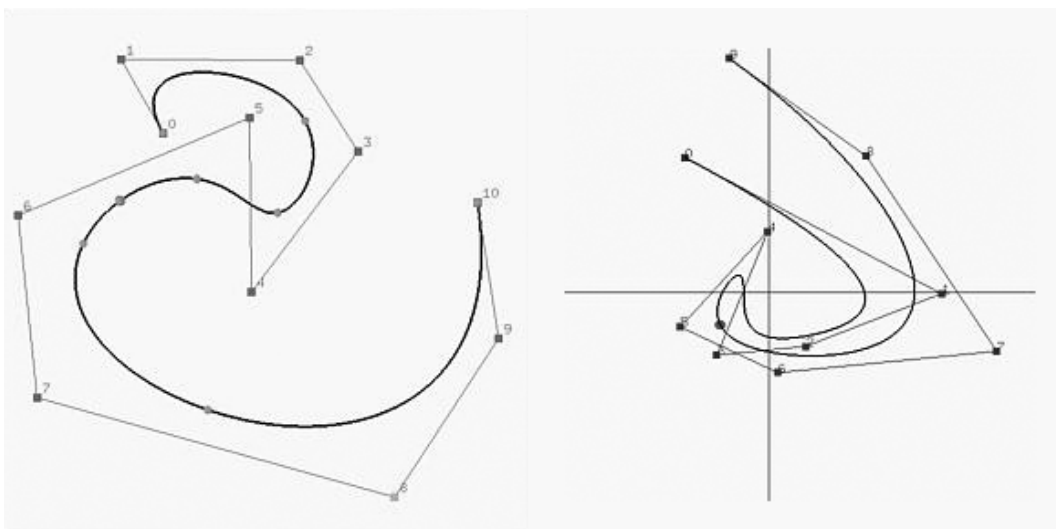


Abb. 13 Eine B-Spline-Kurve und ihre erste Ableitung (Hodograph)

Unter Verwendung von Constraints können benachbarte Kurven-Segmente, die sich

6 Es ist zu vermuten, daß es vom Standpunkt des Nutzers aus vorteilhafter ist, keine „fremden“ rechnergenerierten Striche nachträglich in seiner Zeichnung auftreten zu lassen (wie dies so häufig störend bei Standard-CA(A)D-Anwendungen der Falle ist), um einen möglichst „natürlichen“ Arbeitsablauf – wie im analogen Fall – zu ermöglichen.

innerhalb bestimmter Randbedingungen bewegen, in häufig in der Architektur vorkommende Nachbarschaftsbeziehungen (z.B. Rechtwinkligkeit) „zurechtgerückt“ werden.

#### 4 Von Wänden zu Räumen

Um aus der im vorhergehenden Abschnitt gewonnenen Sammlung von Kurven geschlossene Räume zu extrahieren, müssen in einem ersten Schritt alle Überschneidungen der Kurvensegmente ermittelt werden. Dazu werden alle Kurven einzeln auf mögliche Überschneidungen mit anderen Kurven überprüft, gefundene Kreuzungspunkte markiert und an den entsprechenden Stellen in Untersegmente zerlegt<sup>7</sup>.

Für eine Überführung dieser zunächst rein geometrischen Struktur in topologische Zusammenhänge, wie sie für die Aussage über architekurspezifische Zusammenhänge vonnöten sind, müssen in einem nächsten Schritt Punkte in Knoten und Kurven in Kanten erweitert werden. Dazu werden mithilfe geeigneter Datenstrukturen (*hashtable*, *map*) für alle Punkte jeweils diejenigen Kurven verzeichnet, deren Eckpunkte sie bilden. Umgekehrt werden für ein schnelles Auffinden für jede geometrische Kurve ihr Start- und Endpunkt verzeichnet<sup>8</sup>. Nach dem nun durch geeignete Methoden die „losen Enden“, (also diejenigen Kanten mit Knotenpunkten, die nur von einer einzigen Kante „benutzt“ werden) entfernt wurden können nun durch „Abblauen“ geschlossene Flächen ermittelt werden:

Dazu wird an beiden Endpunkten jeder Kante zunächst ihr jeweiliger Nachbar „im Uhrzeigersinn“ und „gegen den Uhrzeigersinn“ ermittelt. Um diese für gekrümmte Kurvensegmente errechnen zu können, wird ein beliebiger Punkt mit möglichst klein zu haltendem Abstand zum Endpunkt ermittelt, was durch die parametrische Beschreibung der Kurve keine größeren Schwierigkeiten bereitet. Die so gewonnenen Informationen werden in eine Winged-Edge Struktur überführt, wie sie Baumgart in

---

7 Dies geschieht ähnlich den herkömmlichen Arbeitsschritten mit denen Architekten und technische Zeichner in ihren CAD-Anwendungen „Trim“-Funktionen bei sich überkreuzenden Liniensegmenten anwenden. Hier geschieht dies jedoch – wie alle anderen Operationen auch – automatisch ohne jegliches Zutun des Nutzers.

8 Besonderer Berücksichtigung bedürfen dabei „eckenlosen“ gekrümmte Kurvenzüge mit Selbstüberschneidungen, da hier Start- und Endpunkt aufeinander fallen.

[BAUMGART72] vorschlägt. Nun können die Kanten in beiden Richtungen „abgelaufen“ werden. Bis zum jeweiligen Wiedererreichen des Ausgangspunktes bilden alle „unterwegs“ abgelaufenen Kanten einen geschlossenen Polygonzug, der die Grundfläche eines Raumes bildet.

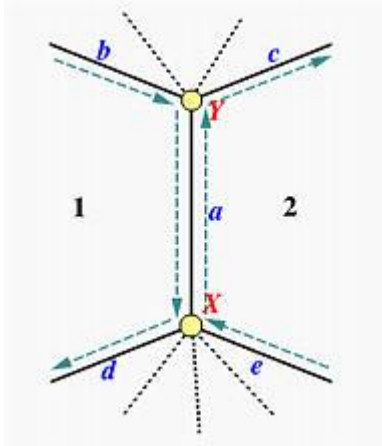


Abb. 14 Schema der Winged-Edge Struktur

Für eine vollständige Beschreibung des architektonischen Grundelementes „Raum“ sind vollständige Ausformulierungen der Wandstärken und ihrer Materialeigenschaften in den frühen Phasen des architektonischen Entwurfes zwar nicht unbedingt von Nöten, wenn diese aber mit „Standard-Werten“ leicht durch das entsprechende Werkzeug erzeugt werden können, ohne daß sich der Anwender mit der zusätzlichen kognitiven Last des Verschneidens, des

Wandaufbaus und ähnlicher Spezifikationen auseinander setzen muß, können sie für verschieden Zwecke auch in dieser Phase hilfreich sein (z.B. „haptische“ Qualitäten unterschiedlicher Wandstärken und Materialitäten, den Lichteinfall durch die Laibungstiefe in Fenstern verändern etc.)

Unter den verschiedenen Varianten des Erzeugens von Wänden als BRep-Repräsentation ist eine der am wenigsten konfliktanfälligen die Erzeugung von Offsets um eine Mittellinie. Der Vorteil, der damit verbunden ist, ist das jeder Raum seine „eigene“ Wand haben kann, was in Bezug auf die Attributierung bspw. von Oberflächeneigenschaften einen niedrigeren Aufwand darstellt als das Verwalten von „zweiseitigen“ Wänden, deren Richtung in Bezug auf angrenzende Räume verwaltet werden muß. Ein dabei entstehender Nachteil ist, daß mit der Wand assoziierte „Features“ wie Öffnungen für jeden Raum einzeln vorgehalten werden müssen<sup>9</sup>

## 5 Von Räumen zum Gebäude

In der Unterscheidung unterschiedlicher Räume ist – nicht nur aus praktischen Erwägungen, wie bauphysikalischen Anforderungen, Arten der Ausbildung oder

<sup>9</sup> Dieser Nachteil wiederum birgt gleichzeitig die Gelegenheit bspw. unterschiedliche Laibungs-Geometrien von Fenstern auf dem außen- oder innenliegenden Teil der Wand zu verwalten.

Brandschutzaufgaben – die Unterteilung in Außen- und Innenräume von großer Wichtigkeit. Außenwände können im topologischen Sinn als diejenigen Kanten definiert werden, die – in Abhängigkeit der Laufrichtung - jeweils nur „rechte“ oder „linke“ Nachbarkanten haben. Im geometrischen Sinn sind Außenwände diejenigen, die begrenzender Teil der größten Fläche aller möglichen Polygone / das größte Volumen aller möglichen Räume haben. Die Gesamtheit aller Räume, ihrer begrenzenden Elemente, sowie ihrer enthaltenen Bauteile machen in diesem Beispiel das gesamte Gebäude aus.

## 6 Türen, Fenster und andere Bauteile

Einzelne Bauteile, wie Türen, Fenster oder Treppen werden in den meisten Handskizzen von Architekten durch Symbole repräsentiert. Beim Übersetzen in eine digitale Repräsentation treten eine Reihe von Schwierigkeiten auf:

1. Die Symbole variieren von Zeichner zu Zeichner
2. Symbole, die aus mehreren einzelnen Strichen zusammengesetzt sind, haben keine feste Erstellungsreihenfolge
3. Die Bedeutung der Symbole ist teilw. abhängig von Größenverhältnissen, Rotation und Position innerhalb des Kontextes (Türaufschläge, Unterschied zwischen Stützen und Pfeiler zu Raumumschließungen etc.)

Ein parametrischer Ansatz hat gegenüber den symbolischen Vorgehensweisen den Vorteil, daß die Informationen, die sich von einzelnen Strichsequenzen gewinnen lassen vielfältig, und nicht an eine feste Reihenfolge der jeweiligen Einzelstriche gebunden sind. Am Beispiel einer Tür läßt sich die Vorgehensweise veranschaulichen: Zunächst können für die Zeichnung einer Tür innerhalb eines Architekturskizze folgende allgemeine Forderungen aufgestellt werden:

1. Eine Wand verbindet immer genau zwei Räume (der Außenraum eines Gebäudes ist lediglich ein Sonderfall eines Raumes)
2. Eine Tür ist Bestandteil genau einer Wand
3. Eine Tür hat einen definierten Aufschlag<sup>10</sup> in genau eine Richtung

Für alle anderen Eigenschaften, wie Art des Blattes, der Zarge oder den jeweiligen

---

<sup>10</sup> Sonderfälle wie Pendel- oder Doppeltüren werden hier vernachlässigt

genauen Abmessungen können jeweils Standardwerte herangezogen werden, die in den frühen Phasen des Entwurfes jedoch eine untergeordnete Rolle spielen.

Auf Grundlage dieser Festlegungen kann nun beispielhaft ein Algorithmus zur Identifikation eines Türelements anhand einer V-förmigen Strichsequenz entwickelt werden:

Zunächst kann anhand der geometrischen Eigenschaften der eingegebenen Strichsequenz ermittelt werden, ob diese einem vorhanden Satz von Regeln entspricht:

Dabei fließen neben allgemeinen geometrischen und topologischen Eigenschaften (Winkel zwischen Strichen, Anzahl der Striche) auch schon maßstabsunabhängige Größen der Elemente ein: Da die Größe von Türen meist innerhalb bestimmter Standard-Parameter variiert (0.76 m, 0.88 m, 1.01 m etc.) werden solche Strichsequenzen, die zwar die anderen Bedingungen erfüllen, aber „architektonisch plausible“ Größen über- oder unterschreiten übergangen.

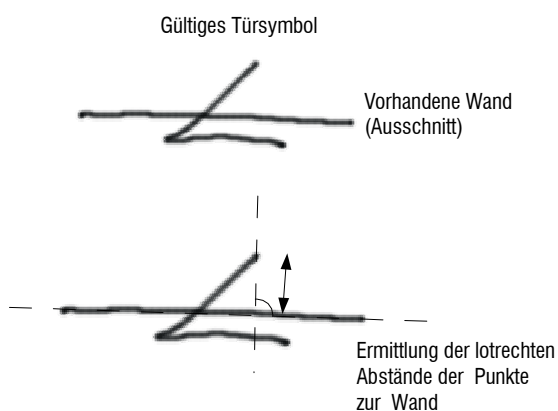


Abb. 15 Ermittlung der Aufschlagsrichtung

In einem nächsten Schritt wird nun die als gültiges Türsymbol validierte Sequenz in Bezug auf ihre Verhältnisse zum Kontext, d.h. bereits vorhandenen Teilen der Zeichnung überprüft.

Dabei muß überprüft werden, ob das Symbol genau eine Wand mindestens einmal mit einer seiner beiden Linien schneidet. (Zur Steigerung der Performanz empfiehlt es sich, zunächst einen groben Intersection-Test gegen das Bounding Rectangle des Symbols mit den einzelnen Raumpolygonen zu machen) Zur Ermittlung der Aufschlagsrichtung kann der am weitesten von der durchschnittenen Wand entfernt

gelegene Punkt herangezogen werden.

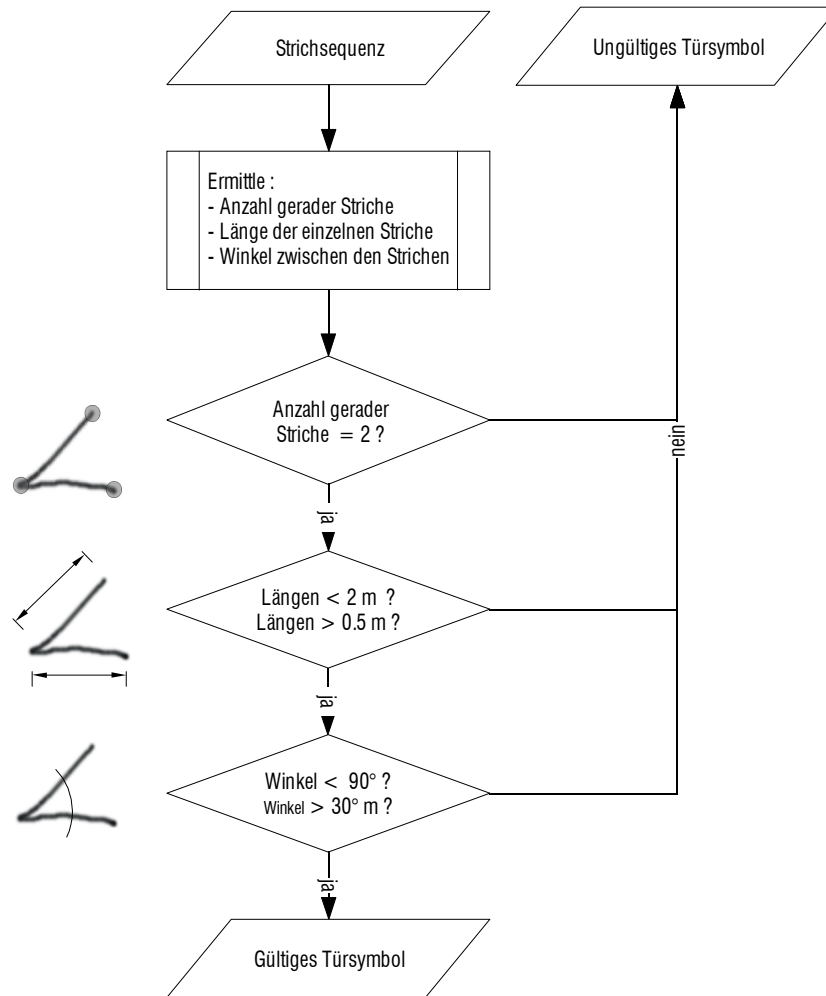


Abb. 16 Entscheidungsablauf zur Erkennung eines Türsymbols

Die beschriebene Methode hat gegenüber anderen Verfahren, wie dem orthogonalen 3x3 Raster den entscheidenden Vorteil, daß alle Symbole in beliebiger Rotation im Raum auftreten können.<sup>11</sup>

Für Symbole, die aus mehreren Einzelstrichen zusammengesetzt sind, kann ein bei jedem neuen Strich aktualisierter Stack zum Einsatz kommen. Bei jedem neuen Strich kann dann eine Sequenz aus den jeweils vorgehenden gebildet werden, und mit bekannten Regeln verglichen werden.

Eine besondere Herausforderung stellen bei jeder Buchstabenerkennung (auch bei der symbolischen 3x3-Grid Erkennung) im Raum gedrehte Varianten dar. Um beliebig in der Ebene rotierte Buchstaben zu erkennen könnte folgende Vorgehensweise

11 Nur so ist es möglich, beliebige Raumgeometrien zu zeichnen, ohne für jede denkbare Rotation eines Symbols einen neuen Regelsatz anzulegen



implementiert werden: Zunächst wird aus der eingehenden Stich- oder Punktwolke das richtungsunabhängige Minimal Bounding Rechteck ermittelt. Aus diesem kann dann ein lokales Koordinatensystem erzeugt werden, dessen Schmalseite die X- und dessen Längsseiten die Y-Achse bilden. Anschließend kann entweder durch eine einfache 3x3 Erkennung oder die oben beschriebene parametrische Auswertung der orthogonale Fall ausgewertet werden.

## 7 Entwurfsunterstützende Zusatzinformationen

Erst nachdem die in den vorangegangenen Abschnitten beschriebene Eingabe der Basisdaten erfolgt ist, kann mit der eigentlichen Entwurfsunterstützung in den frühen Phasen begonnen werden. Nicht alle Informationen, die aus den vorhandenen Angaben ermittelt werden können, sind geeignet das Arbeiten zu erleichtern. Ob eine Information als wichtig oder störend vom Entwerfenden angesehen wird, hängt letztendlich nicht nur von der jeweiligen Art des zu planenden Gebäudes und den individuellen Vorlieben des Architekten ab. Insbesondere die Art der Darbietung kann sich entscheidend auf die Effektivität des Werkzeuges auswirken. So ist kaum damit zu rechnen, daß eine übergroße Anzahl nackter Zahlenkolonnen und Aufmerksamkeit erheischender Einblendungen dem Arbeitsfluß des Zeichnenden zuträglich ist. Vorgeschlagen wird deshalb ein leicht zu bedienendes Interface, daß dem Nutzer jederzeit freie Hand läßt, welche Informationen er zu welchem Zeitpunkt eingeblendet sehen möchte.

Als Metapher dient dabei das aus der „analogen Welt“ jedem Architekten vertraute Transparentpapier: Während des herkömmlichen Entwurfsprozesses werden zur Überprüfung verschiedener Varianten oft mehrere Lagen Transparentpapier z.B. über vorhandenen Planungsunterlagen wie Katasteramts-Pläne oder eigene Vorarbeiten gelegt. Dem Zeichner ist es dabei möglich, zwischen verschiedenen Wahrnehmungen hin- und herzuschalten: Die Zeichnung kann als ganzes, daß heißt als Summe aller einzelnen Lagen betrachtet oder auf die oberste Ebene, die aktuelle bearbeitete Schicht reduziert werden um den jeweiligen Grad der Komplexität der benötigten Informationen zu regulieren. CA(A)D-Programme, bei denen sich diese Ebenen-Steuerung seit langem als allgemeingültiger Standard bewährt hat unterscheiden sich

dabei gegenüber dem physischen Skizzenpapier in einem nicht unwesentlichen Punkt: Ursprünglich als Erleichterung der Organisation von Zeicheninhalten durch Gruppierung angelegt (die meist nur durch mehr aufeinander folgende Befehlssequenzen deaktiviert und aktiviert werden können), hat sich die optische Differenzierung stark vom physikalischen Vorbild entfernt. In den meisten großen C(A)AD-Anwendungspaketen wird eine Unterscheidbarkeit, und damit eine leichtere Lesbarkeit der einzelnen Ebenen meist durch stark kontrastierende Farben der der jeweiligen Ebene zugehörigen Geometrien erreicht. Da es aus historischen Gründen – teilweise noch heute gültige<sup>12</sup> - Beschränkungen des Farbspektrums einer Bildschirmzeichnung gibt, kommen dabei meist Farben zum Einsatz deren Verwendung in einem vom Bildschirmarbeitsplatz losgelösten Kontext aus ästhetischen Gründen wohl eher selten wären<sup>13</sup>. Eine echte optische Tiefenstaffelung des Zeichnungsinhaltes findet, abgesehen von grundlegenden Überdeckungsauswertungen, kaum statt. Dies führt zu einer Gleichwertigkeit aller Elemente<sup>14</sup>. Damit entwurfsunterstützende Einblendungen die Lesbarkeit der Originalskizze nicht beeinträchtigen kann ein Transparenz-Darstellungsmodell, wie es beispielsweise in der Bildbearbeitung verbreitet ist zum Einsatz kommen: Die transluzenten Eigenschaften von Transparentpapier nachempfindend, werden zusätzliche Informationen oder vorhergehende zeichnerische Vorarbeiten je nach Zeichnungshintergrund gegenüber der momentan „obersten“ Ebene heller oder dunkler dargestellt. Ein einzelner Bildpunkt wird dabei mit den darunter- und darüberliegenden Bildpunkten additiv verrechnet. Dadurch entsteht der Effekt des tiefengestaffelten sukzessiven Verblässens von Farben, der – wie beim analogen Vorbild – die Aufmerksamkeit auf der jeweils gültigen Bearbeitungsebene belässt.

Die einzelnen Arbeitsebenen sollten einfach ein- und ausschaltbar sein. Um die

---

12 Auch AutoCad 2002 hat den 255 Farben umfassenden Farbraum trotz durchgängiger 32-Bit-Architektur bis zum heutigen Zeitpunkt noch nicht verlassen

13 Vor allem spielen dabei die Grundfarben des RGB-Farbraumes und ihrer jeweiligen Triplet-Mischungen (110, 101 usw.), die zu exotischen Cyan- und Magenta-Tönen führen, eine Rolle

14 Ausgenommen sei hier das „Einfrieren“ von Ebenen, das eigentlich zum Schutz gegen versehentliche Manipulation der Elemente dieser Ebene gedacht ist, aber in vielen Systemen durch eine gleichzeitige Farbgebung der auf ihr befindlichen Geometrien mit schwachem Kontrast zum Zeichenuntergrund auch für eine optische Abschwächung der entsprechenden Elemente sorgt.

Transparentpapier-Metapher zu stützen, können hervorstehende Teile der jeweiligen Schichten zum Einsatz kommen, wie sie als „Reiter“-Widgets aus vielen Betriebssystemoberflächen bekannt sind. Es ist zu vermuten, daß die unmittelbare optische Kopplung an die eigentliche Ebene auch vorteilhafter ist als die indirekte Steuerung der Sichtbarkeit durch Symbole in separaten (grafischen) Listen<sup>15</sup>. Zusätzlich könnten die grafischen Ebenen frei transformierbar gestaltet werden: Dadurch könnten Zeichnungsteile intuitiv, wie mit dem papierenen Pendant gegeneinander verschoben und rotiert werden, was sich gerade für Entwurfsansätze, die aus konstruktiven Gründen auf Rastern basieren als entscheidende Arbeitserleichterung erweisen könnte. Die Möglichkeit der freien Skalierbarkeit der „Zeichnungsunterlage“ kann in diesen Fällen bei sorgfältiger Konzeption der Schnittstelle ebenfalls zu interessanten positiven Auswirkungen auf den Arbeitsablauf führen.

## 7.1 Grundflächen

Ein häufig in der Praxis anzutreffendes Szenario stellt das Ringen des Architekten mit Quadratmeterzahlen dar: Anforderungen an bestimmte Raumgrößen, die aus Bebauungsplänen, Raumprogrammen, wirtschaftlichen oder gesetzlichen Rahmenbedingungen entspringen, sind gerade in den frühen Phasen des Entwurfes mit steigender Komplexität der Aufgabe schwer zu beherrschen. Was bereits bei rechteckigen Raumgeometrien durch die ständige Zuhilfenahme von Lineal und Taschenrechner zur Ermittlung von Flächengrößen den Arbeitsablauf behindert, wird bei komplizierten Geometrien zum schwerwiegenden Kreativitätshemmnis. Das „Herumzupfen“ und „Schieben“ an den Eckpunkten rechnergenerierter Grundrisse wiederum, das vielfach zur Bewältigung dieses Problems als Werkzeug zum Einsatz kommt, bringt wiederum die mehrfach angesprochenen Problematiken des Mediums mit sich. Auch das nachträgliche Einscannen von Handskizzen und der überschläglichen Flächenermittlung durch Overlay-Techniken stellt einen massiven Eingriff in den flüssigen Arbeitsablauf dar.

---

<sup>15</sup> Siehe Layer-Verwaltung Autocad, Ebenen-Verwaltung Bildverarbeitung The Gimp, Photoshop etc.

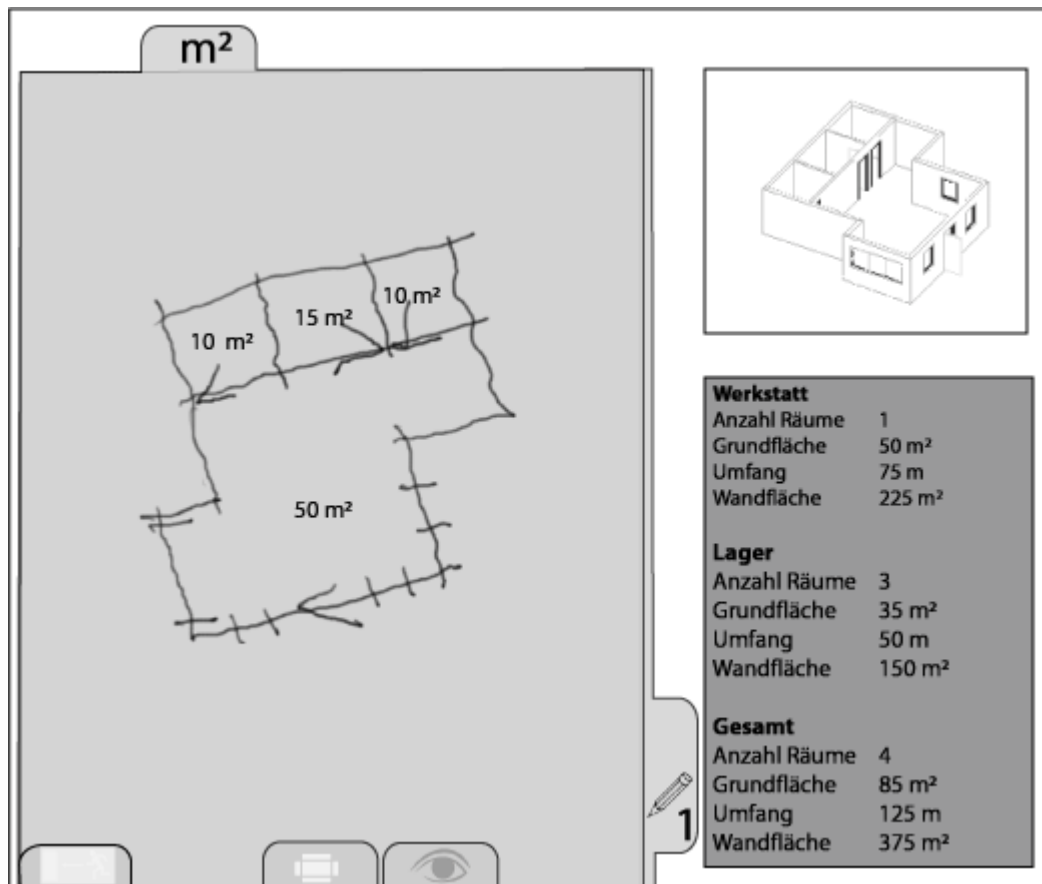


Abb. 17 Einblendung der Raumgrößen

Als erste Zusatzinformation soll hier deshalb die Einblendung von Flächengrößen in die vorhandene Handskizze vorgeschlagen werden. Um den flüchtigen Charakter der Skizze beizubehalten, und weil eine DIN-konforme Mengenermittlung auf ihrer Grundlage kaum möglich ist, sollten angegebene Zahlen überschlägliche, ungefähre Genauigkeiten haben. Eine Angabe „24-26 m²“ oder gerundete ganze Zahlen entsprechen dem Grad der Festlegungsgenauigkeit dieser Phasen des Entwurfes eher als Präzisionen von mehreren Nachkommastellen. Anstelle einer tabellarischen Übersicht, wie sie in anderen Anwendungen nach entsprechender (teilweise auch hier nur nach entsprechender zusätzlicher manueller Festlegung der begrenzenden Geometrien) angeboten wird, sollte von Beginn an eine unmittelbare Zuordnung der Größen durch entsprechende Beschriftung innerhalb der jeweiligen Umfassungsflächen angeboten werden, wie sie aus entsprechenden Planungsunterlagen jedem Architekten vertraut sind. Zusätzlich kann eine Gruppierung in entsprechende Funktionseinheiten durch geeignete Schnittstellen erfolgen, um diese zusätzlich tabellarisch zusammengefaßt darzustellen. Dies kann

durch eine vorher vom Nutzer aufgestellte Liste unterschiedlicher Funktionseinheiten geschehen, die als farblich unterschiedlich markierte Listeneinträge per Drag-and-Drop in die entsprechenden Räume gezogen werden können. Eine andere - wenngleich auch aufwändigere - Methode stellt die direkte handschriftliche Auszeichnung der Räume in der Skizze dar. Dabei können entweder durch einzelne Buchstaben oder durch ganze Worte markierte Räume zu entsprechenden Gruppen zusammengefasst und separat als Übersicht oder abgesetzt von der individuellen Raumbeschriftung innerhalb des Raumes angeboten werden.

## 7.2 Wand- Fußboden- und Deckenoberflächen

Eine weitere Informationsart, die in herkömmlichen C(A)AD-Anwendungen meist nur durch umständliche manuelle Aufbereitung aus den Zeichnungsunterlagen extrahiert werden kann, sind andere überschläglich ermittelte Mengen wie Wandflächen. Ein zentrales Problem stellt dabei die Einpflege entsprechender Informationen bzw. die Auszeichnung der jeweiligen Bauteile dar: Um bspw. eine Wandoberfläche mit einer bestimmten Materialeigenschaft (wie Fliesen, Putz, Wandfarbe) für eine AVA-taugliche Erfassung auszuzeichnen, ist oft die Eingaben von Text oder die ausführliche Beschreibung eines Wandaufbaus nötig. Zudem ist eine visuelle Rückkopplung bspw. in form andersfarbig dargestellter Geometrien in professionellen Anwendungen eher die Seltenheit. Besonders schwierig gestaltet sich dabei die Handhabung von Wänden, auf denen unterschiedliche Oberflächenbehandlungen auf den beiden verschiedenen Seiten vorgesehen ist<sup>16</sup>. Vorgeschlagen wird hier ein Interface, das bei Aktivierung der entsprechenden Zeichenebene eine Abwicklung der Wand-, Boden- und Deckenflächen des entsprechend ausgewählten Raumes einblendet, die durch Drag-and-Drop aus einer grafischen Materialliste ausgezeichnet werden kann. Die so angepassten Informationen zu Raumbooberflächen können dann, nach Materialien sortiert tabellarisch zusammengefasst, als Raumabwicklungen dargestellt und in der dreidimensionalen Präsentation sichtbar gemacht werden. Durch den weiter oben

---

<sup>16</sup> Dies betrifft sowohl Mengenermittlung als auch Visualisierung, in der in den herkömmlichen Lösungen aufwändiges manuelles Per-Face-Mapping erfolgen muß um entsprechend zweiseitige Materialien darzustellen. (z.B. 3DS Max)

beschriebenen Ansatz des zweiseitigen Vorhaltens der Wandgeometrien steht einer unterschiedlichen Auszeichnung von „Vor- und Rückseite“ der Wand nichts im Wege<sup>17</sup>.

### 7.3 Fluchtwege und Brandabschnitte

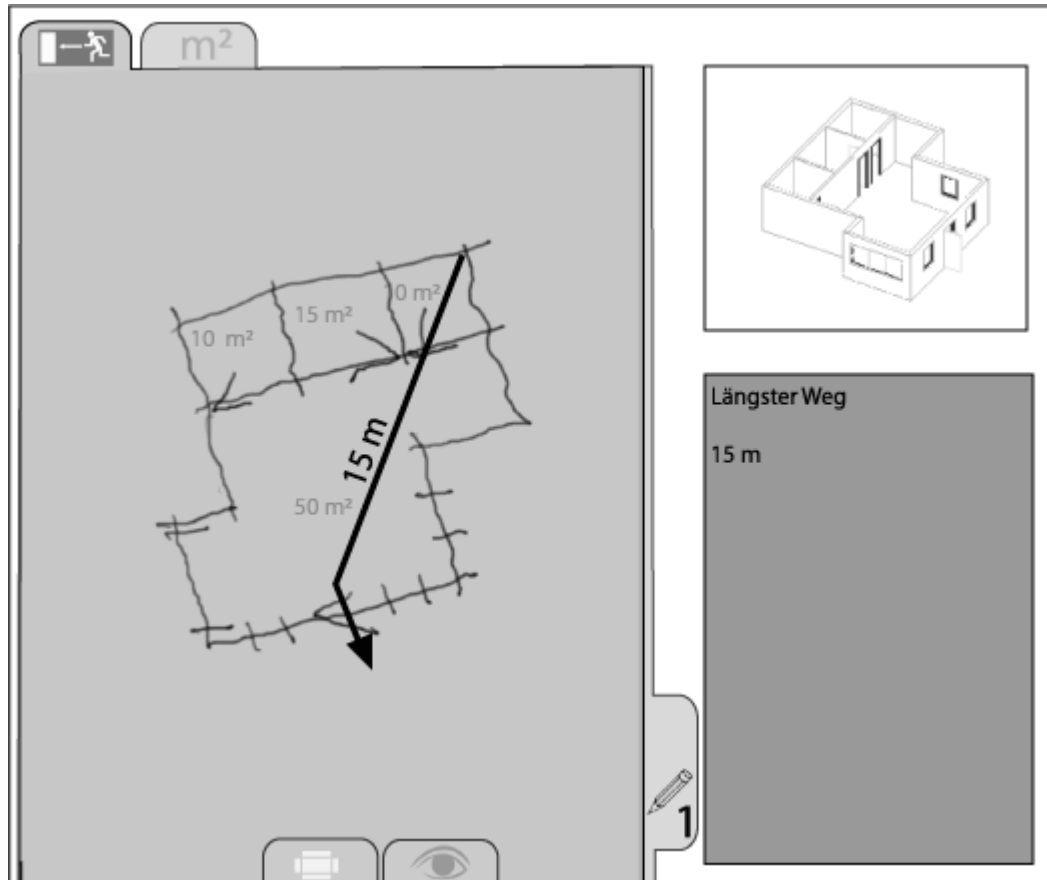


Abb. 18 Eingezeichneter längster Fluchtweg bei aktivierter Fluchtweg-Ebene

Oftmals kritisch in den frühen Phasen des Entwurfs ist die Gliederung eines Gebäudes in separate Brandabschnitte und die Einhaltung entsprechender Rettungsweglängen. Während einfache topologische Mißstände wie gefangene Räume vom Planer selbst erfasst und entsprechend vermieden werden können, stellen beispielsweise die Ermittlung des ungünstigsten Rettungsweges und seiner Länge eine weitere Unterbrechung des Arbeitsflusses dar. Anhand der Raumgeometrien und der Lage der entsprechenden Öffnungen kann dieser in einem hier vorgeschlagenen System „automatisch“ im Hintergrund errechnet und bei entsprechend aktivierter Ebene als Warnhinweis eingeblendet werden.

17 Schwierigkeiten, die bei den entsprechenden Übergangstellen - etwa in Öffnungen - entstehen, können durch die separate Vergabe von Oberflächeneigenschaften der jeweiligen Laibungen oder Zargen umgangen werden.

## 7.4 Ökologische Informationen

Wesentlichen Einfluß auf die Qualität eines Gebäudes haben die Konsequenzen, die sich aus einfachen geometrischen Festlegungen in den ersten Phasen des Entwurfes für die Gesamtenergiebilanz ergeben: Angefangen von grundlegenden Kubatureigenschaften wie dem A/V-Verhältnis über die Zonierung beheizter und unbeheizter Räume bis hin zur Orientierung, Art und Größe der Öffnungen, ist die Gefahr hoch, zu Beginn eines Entwurfes ungünstige Entscheidungen zu treffen, die sich anschließend nur durch teure technische Ausbauten oder die Verwendung entsprechend hochwertiger Materialien (z.B. Gläser oder Dämmungen) ausgleichen lassen. Nicht nur bedingt durch die WSVO, die dem sommerlichen und winterlichen Wärmeschutz auch für Wirtschaftlichkeitsberechnungen zunehmend bedeutend macht, muß gerade diesem Bereich große planerische Sorgfalt gewidmet werden. Zwar gilt auch hier, wie bei anderen zuvor genannten Punkten, daß jeder Planer gewisse Faustregeln und Prinzipien verinnerlicht hat, die Komplexität der beteiligten Faktoren verhindert jedoch oft eine entsprechend ausführliche Berücksichtigung „in Echtzeit“ während des Entwerfens. Da viele der Eingangsgrößen für die Berechnung des Wärmeschutzes rein geometrisch-topologischer Natur sind, eignet sich das hier vorgeschlagene System besonders für diese Aspekte. Andere Größen, vor allem solche, die die detaillierte Materialwahl betreffen (k- und g-Werte) können bspw. aus dem zuvor beschriebenen Werkzeug zur Oberflächenzuordnung gewonnen werden. Bereits ohne daß eine Einnordung des Gebäudes vorgenommen werden muß, kann die Kubatur und das entsprechende A/V-Verhältnis berechnet werden<sup>18</sup>. Die Einblendung von Begradigungen ungünstiger Versprünge in der Fassade könnte als grobe Orientierung dienen, schränkt die gestalterische Freiheit jedoch evtl. zu stark ein. Für eine ausführlichere Bewertung muß dem Nutzer zunächst die Einnordung ermöglicht werden. Dies kann durch die Erkennung gängiger Symbole in den als Außenraumfläche erkannten Teilen der Zeichnung erreicht werden. Anschliessend können die eingezeichneten Öffnungen nach Himmelsrichtung tabellarisch angezeigt oder Einzelwerte an den jeweiligen Bauteilen einblendet werden. Interaktive

---

<sup>18</sup> Dies ist besonders einfach in der Geschoßbauweise mit Flachdach. Komplizierte Dachgeometrien erschweren die Ermittlung zwar, machen diese aber nicht unmöglich

ganzjährige Sonnenstandsimulationen mit entsprechender Anzeige von Sonnen- und/oder Wärmeeinstrahlung in den Innenräumen könnten ein weiteres wertvolles Planungswerkzeug sein. Bei diesen Angeboten sollte jedoch darauf geachtet werden, daß alle erforderlichen Berechnungen in Echtzeit erfolgen können, ohne Auswirkungen auf die (subjektiv empfundene) Reaktionszeit der Anwendung auf Handlungen des Nutzers zu haben<sup>19</sup>. Auch wenn aus Performanzgründen aufwändigere Berechnungen, wie thermodynamische Auswirkungen und physikalisch exakte Wärmebrückenberechnungen mittelfristig nicht realisierbar erscheinen kann auf diesem Sektor die Effektivität architektonischen Entwerfens langfristig gesteigert werden.

### 7.5 Räumliche Eindrücke

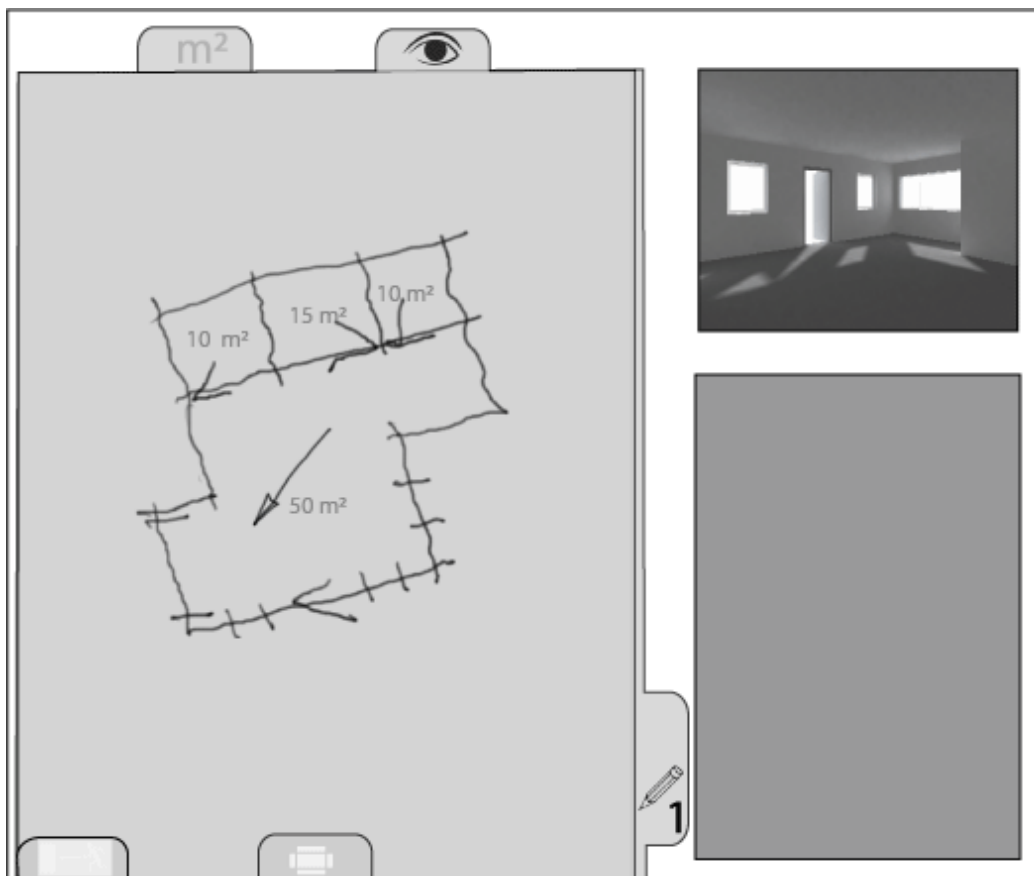


Abb. 19 Durch den auf der aktiven Perspektiven-Ebene eingezeichneten Pfeil wird eine Innenraumperspektive erzeugt

Zur Überprüfung von Raumsituationen durch den Entwerfenden, besonders aber zur schnellen Veranschaulichung für den Bauherren während einer Besprechung, kann

<sup>19</sup> Wie auch bei anderen Berechnungen kann hier evtl eine stufenweise verfeinerte, niedrig priorisierte Hintergrundberechnung in den Idle-Zeiten der Applikation eingesetzt werden.



die Erzeugung von Innen- und Außenraumperspektiven, die in herkömmlichen rechnergestützten Vorgehensweisen meist sehr zeitraubend ist, durch das vorgeschlagene System erleichtert werden. Auf einer separaten Ebenen können dazu schnell entsprechende Blickachsen eingezeichnet werden, worauf in einem gesonderten Bereich der Anwendung das jeweilige perspektivische Bild erscheint. Als gewinnbringend könnte sich auch die automatische Generierung dieser Darstellungen herausstellen: Dabei könnten von jedem Türstock aus automatisch Blicke in jeden Raum erstellt und in einer Galerie zusammengefasst werden.

### 7.6 Zusammenhänge im technischen Ausbau

Im Geschossbau von besonderer Bedeutung ist die Lage einzelner Bauteile für den technischen Ausbau, etwa im Heizungs- und Sanitärbereich. Die genaue Lage etwa eines Schachtes in den darunter- und darüberliegenden Geschossen könnte entweder durch eine entsprechend zurückgenommene Darstellung der jeweiligen Geschosse oder durch die gesonderte Einblendung der erkannten Symboliken (halb diagonal geschwärzter Kasten = Schacht) erfolgen.

### 7.7 Erschließung

Bei der frühen Grobfestlegung einzelner Raumgrößen spielen im Erschließungsbereich Treppen und Aufzüge eine besondere Rolle, da ihre Lage bestimmend für alle Geschosse ist. Während in der täglichen Praxis mit herkömmlichen Paketen die vertikale Erschließung oft durch kopieren einzelner Geschosse geplant wird, gibt es kaum Ansätze für ein durchgängiges, geschoßunabhängiges „Erschließungsrückrat“. Da in den anfänglichen Stadien des Entwurfes oft nur grobe Werte möglicher Treppenformen benötigt werden, um verschiedene Erschließungsvarianten durchzuspielen, könnte die Einblendung einfacher Fälle in Bezug auf die Geschosshöhe hilfreich sein: Gängige Lösungen (ein- zweiläufige Treppenkonstruktion, U-förmig um Fahrstuhlschacht etc.) könnten aus einer Bibliothek eingefügt werden, ohne das dem Nutzer genaue Ausbildungen von Steigungsverhältnis, Wangenform etc. abverlangt werden.

### 7.8 Einbauten und Möblierung

Ebenfalls aus einer Bibliothek können dem Nutzer Standardmöblierungen oder deren

Kubaturen zur Entwurfszeit zur Verfügung gestellt werden. Der Griff zu Werken wie der „Bauentwurfslehre“ von Neufert, dem Architekten in vielen Phasen des Entwurfes Standardabmessungen entnehmen, könnte damit seltener notwendig werden.

# V Implementierung des Prototypen

## 1 Einleitung

Die hier vorgestellte prototypische Implementierung einiger der in den vorangegangenen Abschnitten erörterten Prinzipien ist – der prototypischen Natur und des zeitlichen Entstehungsrahmens gemäß – weit von einem vollständigen, in der täglichen Praxis verwendbaren Werkzeug entfernt. Sie dient vielmehr der Veranschaulichung und Überprüfung einiger Ansätze, die in der Architekturinformatik so noch nicht zum Einsatz gekommen sind.

Die zentrale Rolle, die OpenSource Bibliotheken bei der Umsetzung gespielt haben tragen darüber hinaus zusätzlich zum experimentellen Charakter bei. Die frühzeitig bewußt getroffene Entscheidung, sich nicht auf kommerzielle Werkzeuge zu stützen, hat vielfältige Gründe:

- Finanzieller Aufwand: Allein eine Entwickler-Lizenz für den ACIS<sup>20</sup>-Modeler Kernel kostet um die 20.000 €<sup>21</sup>. Zusätzliche Lizenzierungs-Kosten fallen an, wollte man die damit entstandenen Applikationen anderen zur Verfügung stellen. Obwohl durchaus in anderen Arbeiten verwendet([PICCOLETT098]), ist für den Autoren allein dies ein hinreichendes Ausschlußkriterium für den Einsatz in akademischen Umgebungen.
- Wissenszuwachs: Die Möglichkeit, die einer Bibliothek zugrunde liegenden Quelltexte lesen zu können, stellt nicht nur eine erhebliche Vereinfachung bei der Fehlersuche in eigenen Programmen dar. Sie ist auch eine unschätzbare Quelle der Erkenntnis was Programmiertechniken und -Paradigmen allgemeiner Natur betrifft. Im speziellen Rahmen dieser Arbeit haben sich darüber hinaus dem Autoren fundamentale mathematische und algorithmische Zusammenhänge beim Studium von Quelltexten Dritter erschlossen. Besondere Erwähnung soll hier ausdrücklich die IRIT-Bibliothek<sup>22</sup> von Gershon Elber et al finden, in die eine große Zahl eigener

---

20 Die von der Firma Spatial Technologies Inc. Vertriebene Sammlung von C++ Programmierwerkzeugen zum erstellen, modifizieren und verwalten von BRep Geometrien kommt in unzähligen Produkten des CAD/CAM/CAE – Marktes zum Einsatz. Einer der prominentesten Vertreter ist dabei sicher die Produktlinie der Firma Autodesk<sup>TM</sup>.

21 Stand 2003

22 <http://www.cs.technion.ac.il/~irit/>

- wissenschaftlicher Veröffentlichungen der Autoren, „traditioneller“ Algorithmen der Computergrafik und jahrelange Entwicklungsarbeit eingeflossen sind.
- Plattformunabhängigkeit: Die zum Einsatz gekommenen Bibliotheken sind durchwegs für den Einsatz auf verschiedenen Plattformen geeignet. Der bewußte Verzicht auf betriebssystem-gebundene Werkzeuge wie MFC<sup>23</sup> nimmt dabei eine Schlüsselrolle ein.
  - Support / Community: Da OpenSource Entwicklungen traditionell auf der Beteiligung und der Rückmeldung einer Vielzahl von Anwendern und Entwicklern basieren spielt Kommunikation – oft im Gegensatz zu „ClosedSource“ – eine zentrale Rolle. Durch zahlreiche direkte Diskussionen mit den Autoren der jeweiligen Bibliotheken, sowie den Zugriff auf die Beiträge Dritter, konnten vielfach Klippen umschifft werden, die andernfalls vielleicht nur unter Verwendung kostenpflichtiger Support-Verträge und Hotlines in den Griff zu bekommen gewesen wären. Dieser Umstand macht auch oft die Tatsache mehr als wett, daß verfügbare Dokumentation und Sekundärliteratur in OpenSource-Projekten teilweise dünn gesät sind.

## 2 Grundlegender Aufbau

Die prototypische Implementierung des vorgestellten Ansatzes basiert auf drei wesentlichen Open-Source-Bibliotheken:

1. wxWindows als plattformunabhängige Klassenbibliothek die eine große Zahl nützlicher Werkzeuge bietet. Zu den wichtigsten, die in „Lea“ Verwendung finden zählen:
  - das Application-Framework
  - das Document-View-Framework
  - das Event-Framework
  - die Basisklasse wxObject
  - zahlreiche plattformunabhängige Hilfsklassen wie wxList, wxIniFile etc.
2. OpenCascade als plattformunabhängiger Geometrie- und Brep-Kernel. Die AIS-

---

<sup>23</sup> Die Microsoft Foundation Classes™ sind eine umfangreiche Sammlung zur objektorientierten Kapselung und Erweiterung der Microsoft Windows (TM) API.

Visualisierungsmodule kommen teilweise für die 3D-Repräsentation zum Einsatz, sollten aber in zukünftigen Versionen durch native OpenGL-Routinen ersetzt werden, da sie nicht sonderlich performant sind.

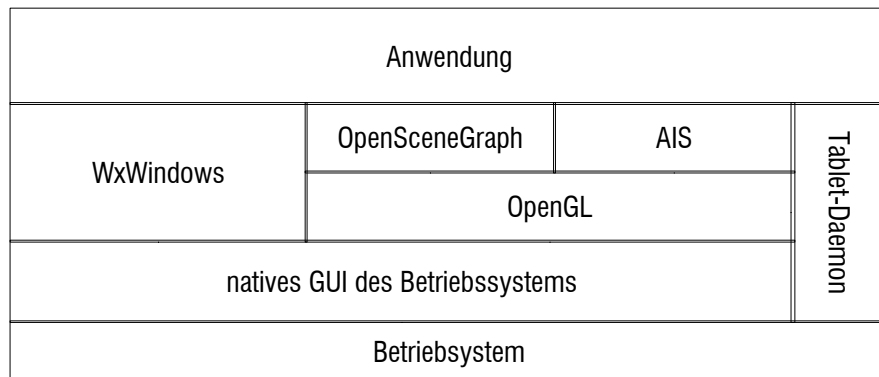


Abb. 20 Übersicht über den Aufbau von Lea

### 3. OpenSceneGraph als plattformunabhängiges Framework für die Visualisierung des Zeichenbereichs, der Ebenenstruktur und der Zeichnungen

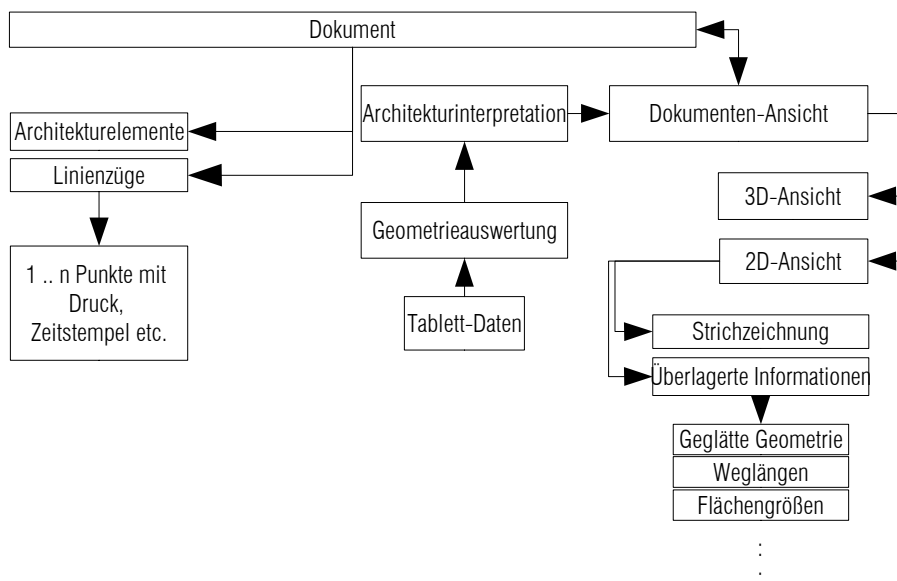


Abb. 21 Übersicht über die Organisation der Programm-Module

Kernstück der Anwendung ist die View-Klasse, von der aus die Erzeugung, Verwaltung und Darstellung der Dokumente zentral gesteuert wird. Ein (nicht plattformunabhängiger) Daemon, der aus allen API-Nachrichten des zugrundeliegenden Betriebssystems Datenpakete des Tablett herauf filtert, erzeugt für jeden vom Tablett übermittelten Punkt einen neuen Event, der an die View-Klasse geschickt wird. Die Instanzen der von wxEvent abgeleiteten Klasse wxTabletEvent

enthalten die folgenden Daten:

| <b>Daten-</b> | <b>Variablenname</b> | <b>Beschreibung</b>   |
|---------------|----------------------|---|
|               | <b>Typ</b>           |   |
| wxPoint       | p                    | /* ganzzahlige Bildschirmkoordinaten in x- und y-Richtung */  |
| wxInt32       | z                    | /* Höhe der Stiftspitze über dem Tablett oder Wert des integrierten Rades (nur gültig bei entsprechender Unterstützung durch die Hardware) */ |
| wxInt32       | pressure             | /* Andruckstärke, Hardware-unabhängig diskrete Stufen von 0 bis 255 */  |
| time_t        | timestamp            | /* Zeitpunkt der Erfassung des Punktes */   |
| wxInt32       | button               | /* Konstante der zum Zeitpunkt der Erfassung gedrückten Knöpfe (z.B. TIP, ERASER, BTN1, BTN2) */  |
| wxInt32       | orAltitude           | /* Rohdaten z-Koordinate oder Radwert */  |
| wxInt32       | orAzimuth            | /* Anstellwinkel relativ zur xy-Ebene */  |
| wxInt32       | orTwist              | /* Rotation um die Längsachse */  |

Bei Empfang eines neuen wxTabletEvents werden die einzelnen Punkte der Eingangsdaten zunächst an die OpenSceneGraph-Visualisierungsklassen gesandt. Hier wird entsprechend der zusätzlichen Informationen (Druck, Geschwindigkeit, Anstellwinkel etc.) für die jeweils letzten beiden eingegangenen Punkte der wxTabletEvents ein neues Liniensegment als GLine dem aktuellen Zeichenlayer der Klasse wxOsgLayer2dNode hinzugefügt. Alle Punkte, die zwischen dem An- und Absetzen des Stiftes auf dem Tablett erfasst worden sind, werden bei Beendigung des Striches zwischengespeichert und durchlaufen dann verschiedene Auswertungen. Dazu wird zunächst von einer Instanz der Klasse wxGeomComputer, die jedem View-Objekt zugeordnet ist, eine approximierende B-Spline-Kurve erzeugt. Aus den in der Kurvendiskussion errechneten Ecken und Wendepunkte werden neue Liniensegmente als Geraden konstruiert. In Abhängigkeit des aktuellen Layers, und damit des vom Nutzer gewählten Modus (Zeichnen, Beschriften etc.) durchlaufen die Geradensegmente anschliessend eine Auswertung. Dabei werden die entsprechenden Charakteristika (Länge, Winkel, Anzahl der Segmente usw.) mit voreingestellten Vergleichswerten verglichen<sup>24</sup>. Je nachdem, was die Analyse ergab, werden die ausgewerteten Segmente an die entsprechenden Klassen und Funktionen

---

24 Die Vergleichswerte, wie etwa zulässige Längen von Türsymbolen, maximale Abweichungen der Punkte von einer approximierten Kurve etc. können in entsprechenden Preferences-Dateien verändert werden.

weitergereicht:

Handelt es sich um Wandlinien werden alle in der Zeichnung befindlichen Liniensegmente durch eine Instanz der Klassen wxOCCGeomComputer miteinander verschnitten (`collectIntersectionCurves ()`) und an eine Instanz der Klassen wxOCCTopo übergeben. Hier wird zunächst ein TopoDS::Wire aus den Segmenten konstruiert (`wxOCCTopo::addWirePoints()`), anschliessend werden die „losen Enden“ des Gitters entfernt (`wxOCCTopo::cutLooseEnds()`) und die verbleibenden Gitterstrukturen in eine Winged-Edge-Struktur überführt. Nach der anschliessenden Konstruktion von Polygonen aus dem Drahtgitter (`wxOCCTopo::extractInnerLoops()`) werden aus diesen Grundflächen in einer angeschlossenen Instanz der Klasse wxArchBuilding entsprechend architektonische Elemente konstruiert und ihre BRep-Strukturen erzeugt (`wxArchBuilding::AddRoom(); wxArchBuilding::addWall()`).

Wurden anhand der Charakteristika der eingehenden Liniensegmente Symbole identifiziert (V-förmige Strukturen als Türen, einzelne Buchstaben zur Beschriftung etc.) werden die Daten an entsprechende Funktionen übergeben, die die resultierenden architektonischen Elemente zu erzeugen versuchen (`wxArchBuilding::addDoor()`) oder andere Arbeitsschritte einleiten.

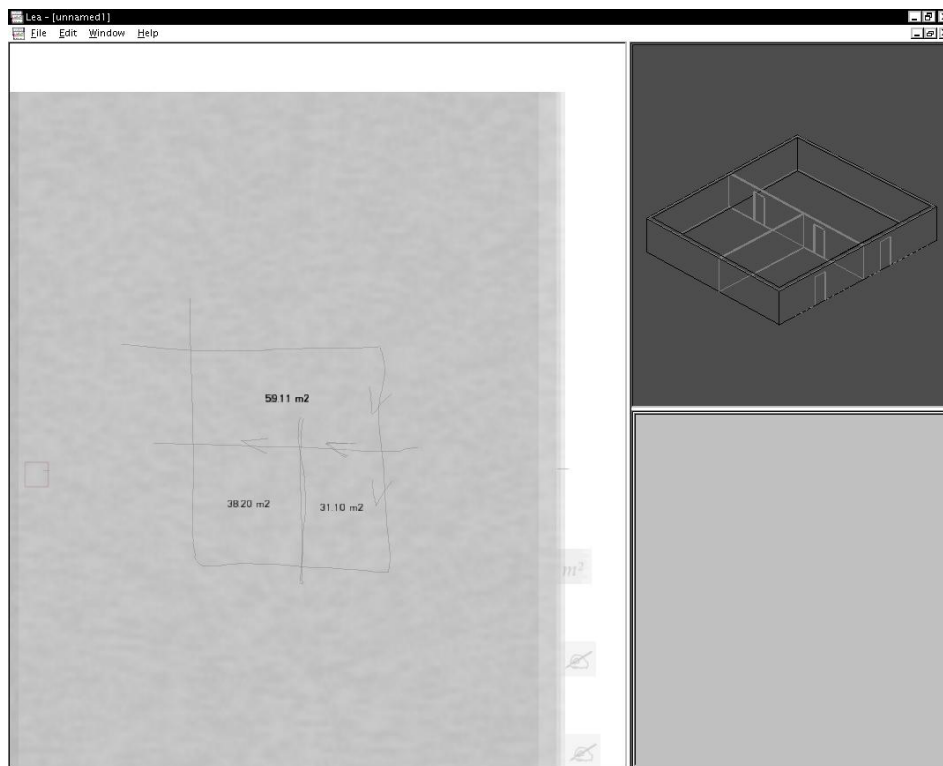


Abb. 22 Screenshot des Prototypen: Im linken Fenster die Handzeichnung mit eingeschalteter Quadratmeteranzeige. In der rechten oberen Ecke eine dreidimensionale Darstellung der interpretierten Geometrie. Die V-förmigen Striche, die durch Wände laufen sind als Türen interpretiert und mittels bool'scher Operationen aus den Wandgeometrien gestanzt



## VI Diskussion und Ausblick

### 1 Zusammenfassung und Diskussion

In der vorliegenden Arbeit wurde ein parametrischer Ansatz für die computergestützte Interpretation von architektonischen Skizzen in den frühen Phasen des Entwurfs vorgestellt. Ausgangspunkt war das Arbeiten mit dem Stift, das auch in einer zunehmend vom Computer geprägten Arbeitsumgebung des Architekten weiterhin eine wichtige Rolle in den frühen Phasen des Architekturentwurfs einnimmt. Ziel der vorgestellten Ansatzes war der Erhalt des flüchtigen, unscharfen Charakters der Handskizze bei gleichzeitiger Erweiterung der in ihr enthaltenen Informationen durch rechnerische Methoden. Zentraler Gegenstand der Betrachtung war dabei auch die Beibehaltung des natürlichen Arbeitsablaufes, wie er aus dem Umgang mit Papier und Bleistift jedem Architekten vertraut ist.

Es wurde gezeigt, daß dieser Ansatz anderen Methoden gegenüber, wie der Analyse einzelner Striche mit Hilfe von 3x3 Rastern ([GROSS96]), eine Reihe von Vorteilen in sich birgt: Der Benutzer muß sich bei der hier vorgestellten Methode weder an bestimmte Reihenfolgen beim Zeichnen der Einzelstriche halten, noch unterliegt er Beschränkungen, die sich aus der Orthogonalität der Rastermethode ergeben. Ein weiterer wesentlicher Vorteil des vorgestellten Verfahrens liegt im Einsatz eines BRep-Kernels, der die Erstellung dreidimensionaler Geometrien und aus ihnen die Gewinnung zahlreicher entwurfsrelevanter Größen ohne weiteres Zutun des Anwenders erlaubt. Die konsequente Beschränkung des Interpretationsrahmens auf spezifische Eigenschaften von Grundrissen (wie beispielsweise bestimmter maßstäblicher Größenverhältnisse einzelner Bauteile zueinander) läßt darüber hinaus eine genauere Deutung der Nutzereingaben zu, als es mit allgemeingültigen Erkennungsverfahren möglich ist..

Zahlreiche Schwierigkeiten, unbeantwortete Fragen und neue Problem werden aus der Arbeit ersichtlich. Dies betrifft nicht nur den Prototypen mit seiner rudimentären Funktionalität, in dem – auch bedingt durch die zeitlichen und technischen Rahmenbedingungen – nur ein winziger Ausschnitt der vorgeschlagenen Ansätze umgesetzt werden konnte. Auch konzeptionell bedarf die Thematik weiterer Arbeit.

So läßt sich beispielsweise lange und ausführlich darüber streiten, wie die Balance zwischen einfacher Handhabung und detaillierter Information herstellbar ist: Auf der einen Seite soll dem Nutzer ein möglichst unsichtbares, natürliches System zur Verfügung gestellt werden, das Zeichnung möglichst „intelligent“ interpretiert. Idealerweise sollte dem Anwender nicht abverlangt werden, seine Handlungen dem System vorher ankündigen zu müssen („Ich möchte jetzt eine Tür in diese Wand einfügen“). Auf der anderen Seite werden die jeweiligen Interpretationen umso genauer, je mehr und je genauer die Festlegungen durch den Benutzer werden. Ein Gleichgewicht beider Teile kann dabei nur unter Berücksichtigung individueller Vorlieben, dem jeweiligen Einsatzzweck und der Art des zu entwerfenden Gebäudes gefunden werden.

## 2 Ausblick

„Natürliche“ Mensch-Maschine-Schnittstellen, wie Spracheingabe, und Handschriftenerkennung spielen in vielen derzeitigen Entwicklungen von Computeranwendungen eine zunehmend große Rolle. Nachdem einige der ersten Kinderkrankheiten und Schwierigkeiten beseitigt werden konnten und benötigte Rechengeschwindigkeiten aufwändiger Algorithmen jetzt oder in naher Zukunft zur Verfügung stehen, können die inhaltlichen Probleme angegangen werden. Dazu zählen im Falle der Architekturapplikationen die Steigerung der Bedienerfreundlichkeit für die Erstellung von Zeichnungen und Plänen, die Herabsetzung der Hürden für die Einpflege zusätzlicher bauwerksrelevanter Informationen durch die verschiedenen Fachplaner (auch hier liegt ein Schwerpunkt auf entsprechenden Interfaces) und schließlich die Etablierung transparenter gemeinsamer Standards für den Datenaustausch. Der in der Arbeit vorgestellte Ansatz, der zugehörige Prototyp, vor allem aber die neu gewonnenen Einblicke und Erkenntnisse werden für den Verfasser der Arbeit Teil der Basis zukünftiger Forschungsarbeit werden. Zu den konkreten Ansatzpunkten, das vorgestellte System zu erweitern zählen:

- Erstellung eines Versionierungssystems: Die Entstehungsgeschichte einer Handskizze könnte in Weiterentwicklungen in jedes beliebige Stadium zurückgesetzt werden. Dabei könnte ein visueller „Entstehungsbaum“ zum Einsatz

kommen, mithilfe dessen der Anwender von jeder beliebigen Stelle aus neue Variationen in Form neuer Äste und Zweige anfertigen könnte.

- Weiterentwicklung der Transparentpapier-Metapher: Ebenen könnten mit der Stiftspitze frei rotier- und verschiebbar gemacht werden<sup>25</sup>. Übereinanderliegende Ebenen ergeben dabei jeweils ein neues Bild (Stützenraster, innerhalb derer Elemente verschoben werden, gegeneinander verdrehte Raster wie in P. Eisenmanns „House III“ etc.)
- Erweiterung für die Nutzung im Bauaufmaß: Da Tablet-PCs aufgrund ihres geringen Gewichtes und ihres kleinen Formates hervorragend für die Arbeit vor Ort geeignet sind könnten Applikationen entstehen, in denen der aufmessende Architekt zunächst eine grobe Skizze des Raumes zeichnet, die dann Schrittweise durch die handschriftliche Ergänzungen wie vor Ort gemessener Werte verfeinert wird, die direkt an die jeweiligen Bauteile angetragen werden. Darüber hinaus stehen durch drahtlose Technologien wie Wireless LAN zahlreiche Möglichkeiten zur Verfügung die Daten der entsprechenden speziellen Aufmaßgeräte direkt in der Applikation zu verarbeiten.
- Integration in kollaborative Umgebungen: Der Prototyp könnte als Werkzeug in kollaborativen Umgebungen [FRÖHLICH97] genutzt, in der es mehreren am Planungsprozeß Beteiligten möglich ist gleichzeitig (gemeinsam vor Ort oder über entsprechende Netzwerke) anhand von Skizzen über verschiedenen Aspekte des Bauprozeßes zu diskutieren.
- Generalisierung des parametrischen Konzeptes für das Training weiterer Symbole: Durch eine geeignete Notation (und zusätzlich evtl. einer Skriptsprache) könnte die Anwendung leicht auf spezielle Bedürfnisse angepasst werden.
- Integration in vorhandene Applikationen: Die entstandene Anwendung könnte als eine Art „Transparentpapier“-Plugin über herkömmlichen CA(A)D-Anwendungen liegen, die die erstellten Geometrien im jeweils nativen Format oder in einem neutralen Standard (IFC, STEP) an die eigentliche Applikation weitergibt, wo sie

---

25 Viele Stifte in Digitalisieretabletts erfassen bereits die Drehung um die Längsachse, so daß natürliche Bewegungsformen von gleichzeitiger Rotation und Translation in der Ebene möglich sind. In den meisten vorhandenen Anwendungen bleibt dieser zusätzliche Freiheitsgrad – ähnlich wie der Anstellwinkel – ungenutzt.

verarbeitet werden können.

- Dreidimensionale Erweiterung: Die häufig eingesetzte Darstellung dreidimensionaler Sachverhalte in Form von Isometrien und Militärperspektiven könnte in zukünftigen Entwicklungen nutzbar gemacht werden: Dabei werden wie im implementierten Prototypen zunächst die Grundrisse erzeugt, die dann aus verschiedenen Perspektiven betrachtet werden können. Zeichnerische Manipulationen, die in der isometrischen Ansicht vorgenommen werden (z.B. Einzeichnen von Fensteröffnungen in der Außenperspektive) werden anhand des momentanen Blickwinkels auf die Geometrien projiziert und interpretiert.

## VII Anhang

### 1 Literaturverzeichnis

- ALBERTI12 Leon Battista Alberti, *Zehn Bücher über die Baukunst. Ins Dt. uebertr., eingeleitet und mit Anm. und Zeichn. vers. durch Max Theuer*, Heller; Darmstadt; 1912
- BAUMGART72 Baumgart, Winged Edge Polyhedron Representation, Stanford Artificial Intelligence Laboratory; Stanford University;1972
- BOOR78 Carl de Boor, *A Practical Guide to Splines*, Springer Verlag;; New York 1978
- D001 Ellen Yi-Luen Do, *VR Sketchpad - Create Instant 3D Worlds by Sketching on a Transparent Window*, in: Proceedings of CAAD Futures 2001, 2001
- DONATH88 Dirk Donath, *Untersuchungen zur anwenderspezifischen Kommunikation und Modellierung im computergestützten Architekturoentwurf.*, Hochschule für Architektur und Bauwesen Weimar, 1988
- DUDEN89 Duden, *Deutsches Universalwörterbuch*, Dudenverlag; Mannheim, Wien, Zürich 1989
- EGGLI97 Lynn Egli, Ching yao Hsu, Beat D Bruderlin, and Gershon Elber, *Inferring 3d models from freehand sketches and constraints*, in: Computer-Aided Design, 29(2),1997
- FERGUSON77 E.S. Ferguson, *The mind's eye: nonverbal thought in technology*, Science 197, 1977
- FRÖHLICH97 Bernd Fröhlich, Martin Fischer, Maneesh Agrawala, Andrew Beers, Pat Hanrahan, *Collaborative Production Modeling and Planning*, IEEE Computer Graphics and Applications, Vol 17 Issue 4,1997
- GOLDSTEIN96 Bruce Goldstein, *Sensation and Perception; 6th edition*, Wadsworth; New York 1996
- GROSS96 Mark Gross, Ellen Yi-Luen Do, *Demonstrating the Electronic Cocktail Napkin - a paper-like interface for early design*, in Proceedings of CHI, 1996
- HUTCHINS86 Hutchins, Hollan und Norman, *Direct manipulation interfaces*, in: D.A. Norman und S.W. Draper: *User centered system design*, Lawrence Erlbaum Associates, Inc, Hillsdale,N. J.,1986
- IGARASHI01 Takeo Igarashi, John F. Hughes, *A Suggestive Interface for 3D Drawing*, :in Proceedings of ACM UIST 2001, 2001
- IGARASHI98 Takeo Igarashi, Satoshi Matsuoka, Sachiko Kawachiya, Hidehiko Tanaka, *Pegasus: A Drawing System for Rapid Geometric Design*, CHI'98 Summary, 1998
- JENKINS92 D. L. Jenkins, R. R. Martin, *Applying Constraints to Enforce User's Intentions in Free Hand 2-D Sketches*, Intelligent Systems Engineering, 1992
- LEEUWEN99 Jos P. van Leeuwen, *Modelling Architectural Design Information by Features*,

- Eindhoven University of Technology, 1999
- LIEBICH93 T. Liebich, *Wissensbasierter Architekturentwurf - Von den Modellen des Entwurfs zu einer intelligenten Entwurfsunterstützung*, Hochschule für Architektur und Bauwesen, 1993
- MITCHELL94 William J. Mitchell, *The logic of architecture, design, computation and cognition*, MIT Press; Cambridge, Mass. 1994
- MYERS98 Brad A. Myers, *A Brief History of Human Computer Interaction Technology*, ACM interactions 1998, 1998
- MYNATT99 Elizabeth Mynatt, Takeo Igarashi, Keith Edwards and Anthony LaMarca, *Flatland: NewDimensions in Office Whiteboards*, in Proceedings of CHI`99, 1999
- OECHSLIN02 Werner Oechslin, *Lineamenta - Liniengefüge*, unveröffentlichtes Typoskript, 2002
- PALLADIO93 Andrea Palladio, *Die vier Buecher zur Architektur Nach der Ausgabe Venedig 1570 I quattro libri dell'architettura aus dem Italienischen uebertragen und herausgegeben von Andreas Beyer*, Artemis Verlag; Zürich 1993
- PETZOLD01 Frank Petzold, *Computergestützte Bauaufnahme als Grundlage für die Planung im Bestand*, Bauhaus-Universität Weimar, 2001
- PICCOLETT098 Moreno A. Piccoletto, *Sketchpad+ Architectural Modeling Through Perspective Sketching on a Pen-Based Display*, Cornell University, 1998
- REGENBRECHT00 Regenbrecht, H., Kruijff, E., Donath, D., Seichter, H. and Beetz, J., *VRAM - A Virtual Reality Aided Modeller*, in Proceedings of 18th eCAADe, 2000
- RUDISILL95 M. Rudisill, C. Lewis, P. Polson and T. McKay (Hrsg), *Human-Computer Interface Design: Success Stories, Emerging Methods, and Real-World Context*, Morgan Kaufmann; 1995
- SCHMITT93 Geerhard Schmitt, *Architectura et machina - Computer Aided Architectural Design und virtuelle Architektur*, Vieweg; Braunschweig 1993
- SCHMITT96 Gerhard Schmitt, *Architektur mit dem Computer*, Vieweg; Braunschweig 1996
- SEICHTER02 Hartmut Seichter, *Die architektonische Skizze in Virtuellen Umgebungen*, Bauhaus Universität Weimar, 2002
- STACHOWIAK73 H. Stachowiak, *Allgemeine Modelltheorie*, Springer Verlag; Wien 1973
- SUTHERLAND63 Ivan E. Sutherland, *SketchPad: A Man-Machine Graphical Communication System*, in Proceedings of n AFIPS Spring Joint Computer Conference, 1963
- VRIES00 B. de Vries, *Sketching in 3D*, in Proceedings of 18th eCAADe, 2000

## 2 Abbildungsverzeichnis

|  |    |
|--|----|
| Abb. 1 Das Whiteboard von Flatland   | 14 |
| Abb. 2 Vier Beispiel-Verhaltensweisen in Flatland  | 15 |
| Abb. 3 Interface- Metaphern in Flatland  | 15 |
| Abb. 4 Quick-Sketch: "Zittrige" Eingangsskizzen  | 17 |
| Abb. 5 Quick-Sketch Bereinigte Eingangsskizzen   | 17 |
| Abb. 6 Quick-Sketch: "gesäuberte" Eingangsskizze mit Darstellung der verwendeten Constraint.<br>Rechts oben die eingezeichnete Extrusionsachse | 18 |
| Abb. 7 Quick-Sketch: perspektivische eingezeichnete Features   | 18 |
| Abb. 8 Cocktail Napkin: Das Interface des Prototypen   | 19 |
| Abb. 9 Cocktail Napkin: Konsole für die Symbolsprache zum Erstellen und vergleichen von Gesten   | 19 |
| Abb. 10 Konfigurationsdialoge für die Erstellung einer Tür im AutoCad Architectural Desktop  | 22 |
| Abb. 11 Tablett-PC (Compaq TC 1000)  | 24 |
| Abb. 12 Ungeglättete und geglättete approximierte Spline-Kurve mit Gewichten   | 27 |
| Abb. 13 Eine B-Spline-Kurve und ihre erste Ableitung (Hodograph)   | 27 |
| Abb. 14 Schema der Winged-Edge Struktur  | 29 |
| Abb. 15 Ermittlung der Aufschlagsrichtung  | 31 |
| Abb. 16 Entscheidungsablauf zur Erkennung eines Türsymbols   | 32 |
| Abb. 17 Einblendung der Raumgrößen   | 36 |
| Abb. 18 Eingezeichneter längster Fluchtweg bei aktivierter Fluchtweg-Ebene   | 38 |
| Abb. 19 Durch den auf der aktiven Perspektiven-Ebene eingezeichneten Pfeil wird eine<br>Innenraumperspektive erzeugt                           | 40 |
| Abb. 20 Übersicht über den Aufbau von Lea  | 45 |
| Abb. 21 Übersicht über die Organisation der Programm-Module  | 45 |
| Abb. 22 Screenshot des Prototypen  | 48 |

### 3 Danksagung

Diese Abschlußarbeit wäre ohne Einflüsse und Unterstützungen zahlreicher Menschen nicht möglich gewesen. Besonderer Dank gebührt Prof. Donath für die jahrelange akademische Betreuung und die Ermöglichung zahlreicher „Ausflüge“. Weiterer Dank gilt: Holger Regenbrecht, der mir Initialzündung, Inspiration und Mentor war, Frank. Petzold für viele anregende Gespräche und Bernd Fröhlich der mir zahlreiche Einblicke in VR und Computergraphik ermöglicht hat. Allen Mitgliedern der igroup danke ich für die vielen Jahre inspirierender, kontroverser und kreativer Diskussionen und Projekte. Den Mitarbeitern des Lehrstuhl InfAR gebührt für den fachlichen Austausch und die großartige Arbeitsatmosphäre großer Dank. Die Software hätte ohne die Hilfsbereitschaft der OpenSource-Gemeinschaft nicht entstehen können. Neben unzähligen anonymen „Bazaar-members“ gilt einigen Personen besonderer Dank: Vadim Zeitlin, Julian Smart (wxWindows); Robert Osfield (OSG); Gershon Elber (IRIT), all authors of OpenCASCADE, thank you for your amazing source and your support!

Danken möchte ich meiner Familie, die mir das Studium ermöglicht hat und mir über viele Jahre jede erdenkliche Form der Unterstützung zukommen ließ. Anke danke ich für die Geduld, den moralischen Rückhalt und die geisteswissenschaftlichen Ratschläge und Ermahnungen. Hendrik, danke für den Rechenknecht (adopt-a-geek), Hartmut, Dank für wxICQManual usw., Freunde, danke für den Analogweltanker.