Hochschule für Angewandte Wissenschaften Hamburg
Fachbereich Elektrotechnik und Informatik

Studienarbeit

Konzeption eines Systems zur PC-Bedienung mittels Gestenerkennung im Sinne des 'Disappearing Computers'

Erstellt im Studiengang Softwaretechnik
von Martin Senkbeil, Matr.-Nr.: 1562967

Betreuer : Prof. Dr. rer. nat. Kai von Luck

Inhalt

1 Einleitung

Um das eigentliche Ziel und den Ansporn zu dieser Arbeit zu verstehen, sollte zuerst einmal geklärt werden, was es mit dem Ausdruck "Disappearing Computer" auf sich hat.
Er steht für eine Vision, für die [Streiz-05] eine gute Umschreibung liefert: *"Computers became primary objects of our attention resulting in an area called "human computer interaction." Today, however, we must ask: Are we actually interested in interacting with computers? Isn't our goal rather to interact with information, to communicate and to collaborate with people? Shouldn't the computer move into the background and disappear?"*

Sie baut dabei auf der Vision des "Ubiquitous Computing" auf, welche 1991 von Mark Weiser definiert wurde [Weiser-91]. Seine Vision sah es vor, dass wir in der Zukunft von vielen Rechnern umgeben sein werden, welche nahtlos in unsere Umgebung integriert sein werden und über für uns transparente Interfaces verfügen. Wir werden sie kaum noch bewußt wahrnehmen und so natürlich und intuitiv benutzen, wie andere alltägliche Gegenstände.

Das Ziel des "Disappearing Computers" ist es daher, dass die Computer wie wir sie heute kennen immer mehr in den Hintergrund treten, damit die Benutzer nicht primär damit interagieren müssen. Sie sollen vielmehr den Menschen nur bei seinen Tätigkeiten unterstützen statt vorrangig dessen Aufmerksamkeit an sich zu binden. Benutzer wären dadurch in der Lage sich auf das konzentrieren zu können, was sie eigentlich tun wollen. Nicht wie heutzutage, wo oft ein großer Teil der Konzentration allein schon von dem Bedienen der Computer beansprucht wird. Der Computer soll quasi in den natürlichen Kommunikationsapparat des Menschen integriert werden.

Damit jedoch die direkte Interaktion mit der Maschine immer mehr in den Hintergrund treten kann, müssen die Rechner intelligenter und autonomer werden, und es gilt alternative, intuitivere Bedienungsmöglichkeiten zu finden. Hinzu kommt auch die zunehmende Miniaturisieren, die jetzt schon zu immer kleineren und spezialisierteren Computern führt. Ein Trend, der sich in Zukunft noch weiter fortsetzen wird und ebenso alternative Formen der Interaktion erforderlich macht.

Stellenweise hat sich ja auch schon einiges getan. So gibt es heute bereits einige Alternativen, die die Bedienung von Computern verbessern:

- Touchscreens, die es erlauben intuitiv mit den Fingern auf Knöpfe und Menüpunkte zu drücken anstatt umständlich einen Zeiger mit der Maus bewegen zu müssen.
- PDAs, welche die Eingabe von Text in handschriftlicher Form ermöglichen und somit das umständliche Bedienen einer Tastatur ersetzen.
- Graphic-Tablets, mit denen man am Computer auf natürliche Weise mit Stift und Lineal zeichnen kann.

Die heutigen Alternativen bieten zwar schon einen gewissen Grad an Intuitivität, jedoch bleibt für die Integration des Computers in den natürlichen Kommunikationsapparat des Menschen noch einiges zu tun. So sind z. B. Sprache, Mimik und Gesten die natürlichsten Kommunikationsformen für den Menschen und es ist daher wünschenswert, dass sie zukünftig auch für die Interaktion mit Computern verwendet werden können.

Dazu hat z. B. Jakob Nielsen in [Nielsen-93] einige interessante Visionen für Nachfolger WIMP[1]-basierter Betriebssysteme, wie sie auch heute noch verwendet werden, gehabt. Er sah schon damals unter anderem die Verarbeitung von Gesten oder Sprache als Eingabemedien für zukünftige Systeme vor. Bislang hat der Stand der Forschung und Entwicklung allerdings die Verbreitung solcher Systeme noch verhindert und selbst heutzutage sind sie noch nicht zufriedenstellend umzusetzen.
Zwar ist die Technik mittlerweile so weit, dass selbst kleine und preisgünstige Computer über eine hohe Rechenleistung verfügen, aber die notwendigen theoretischen Grundlagen der Erkennung von visuellen oder akustischen Informationen arbeiten noch nicht zuverlässig genug, um unter allen Bedingungen akzeptable Ergebnisse erzielen zu können. Die Forschung hat gerade erst angefangen die komplizierten Vorgänge des Hörens bzw. Sehens zu verstehen.

Das Ziel dieser Arbeit ist es daher lediglich ein Konzept für ein System zu erstellen, welches es ermöglichen soll die Interaktion mit einem Computer per intuitiver Steuerung anhand von Gesten durchzuführen.

Für weiterführende Informationen, Beispiele und Entwicklungen zum Thema "Disappearing Computer" sei hier noch auf [DisappearingComputer-05, Russel-05] verwiesen.

1.1 Überblick über die folgenden Kapitel

Kapitel 2 stellt zwei Visionen dar, in deren Rahmen der Gegenstand dieser Arbeit angewendet werden könnte. In Kapitel 3 wird ein kleiner Überblick über die visuelle Gestenerkennung gegeben und Kapitel 4 widmet sich der Ausarbeitung des Konzeptes für das Bedienungssystem auf Basis von Gesten. Abschließend wird in Kapitel 5 ein Resümee der Arbeit gezogen.

[1] WIMP – Kurzform für "Windows, Icons, Menus, Pointing device". Eine Beschreibung für grafische Benutzeroberflächen, wie sie heutige Betriebssysteme bereitstellen.

2 Visionen

In diesem Kapitel werden zwei Visionen beschrieben, für die der Gegenstand dieser Arbeit verwendet werden könnte.

2.1 System für Präsentationen

Im Sinne des "Disappearing Computers" wäre es z. B. ein interessantes Projekt, wenn ein kleiner "intelligenter" Präsentationsrechner im Vortragspult oder einem anderen Gegenstand eingelassen wird, welcher intuitiv durch Sprache und Gesten bedient wird. Der Ablauf der Bedienung solch eines Systems könnte dabei wie folgt aussehen:

1. Der Moderator betritt den Raum und überträgt per gesprochenem Befehl seine Präsentation von einem PDA, USB-Stick oder aus dem Netz auf das Präsentationssystem.
2. Er startet die Präsentation durch Sprache oder Gesten.
3. Die eigentliche Präsentation wird durchgeführt. Das Durchschalten der Folien, sowie dynamische Hervorhebungen auf selbigen geschieht anhand von Gesten.
4. Der Moderator beendet die Präsentation durch Sprache oder Gesten.
5. Er löscht die übertragene Präsentation vom Präsentationssystem per gesprochenem Befehl. Alternativ wird sie beim Abschalten des Systems automatisch wieder gelöscht.

Die Gesten- und Spracherkennung würden in solch einem System eine sehr gute und intuitivere Alternative für die Bedienung darstellen, denn der Vortragende müßte sich nicht mehr direkt mit der Technik bzw. dem PC befassen und könnte sich ganz auf seine Präsentation konzentrieren.
Das System müßte bei diesem Szenario zusätzlich über eine gewisse "Intelligenz" verfügen. Es muß kontextabhängig feststellen können, ob erkannte Befehle auch wirklich vom Moderator gewünscht werden, oder ob sie nur Teil seines Vortrages sind.

2.2 Home-Multimedia-Center

Im diesem Bereich wäre es denkbar, dass ein heimisches Multimedia-Center eingerichtet und mit Hilfe von Gesten bedient wird. Dadurch könnte der Benutzer dann entspannt im Sessel sitzen bleiben während er mittels Gesten z. B. die Lautstärke reguliert, oder den Fernsehkanal wechselt. Es wäre dann bei der heutigen Vielzahl von Geräten (CD, DVD, TV, ...) nicht mehr notwendig sich mit diversen - oder wenigen komplizierten - Fernbedienungen beschäftigen zu müssen.

Die Gestenerkennung eignet sich für solch ein System sehr gut als alternative Form der Bedienung, da Spracherkennung aufgrund der Hintergrundgeräusche bei laufender Musik etc. mitunter nur sehr schlechte Ergebnisse erzielen kann. Der heutige Stand der Technik liefert im Bereich der Spracherkennung leider noch keine ausreichende Qualität, aber in naher Zukunft wird es bestimmt so weit sein.

3 Allgemeines zur visuellen Gestenerkennung

Die visuelle Gestenerkennung basiert auf der Bildverarbeitung und stellt einen Bereich aus dem Umfeld 'Computer Vision'[2] dar. Ihr Ziel ist es, durch Verfolgung von Bewegungen die Durchführung von Gesten zu erkennen. Verfolgt werden dabei ausgezeichnete Marker, bei denen es sich prinzipiell um beliebige Objekte bzw. deren Eigenschaften handelt.

Nachfolgend werden in diesem Kapitel die wesentlichen Merkmale zur Erkennung, sowie generelle Schwierigkeiten, mit denen man konfrontiert wird, vorgestellt.

3.1 Merkmale der Erkennung

Wie bereits erwähnt werden Gesten durch die Verfolgung bestimmter Objekte oder Eigenschaften erkannt. Darauf soll hier ein wenig genauer eingegangen werden, indem wesentliche Merkmale angeführt werden, die im allgemeinen für die Bewegungserkennung herangezogen werden.
In der Praxis werden diese oft in Kombination miteinander verwendet. Ein Beispiel dafür stellt das Projekt von [Siebel-03, Siebel-05] dar, das sich der Erkennung und Verfolgung von Personen in Videobildern widmet.

3.1.1 Farben

Den einfachsten zu erkennenden und zu verfolgenden Marker bzw. die einfachste Eigenschaft stellen Farben dar, denn jedes einzelne Pixel[3] eines digitalen Bildes wird direkt mit seiner Farbe beschrieben. Daher kann selbige einfach ausgelesen und z. B. deren Schwerpunkt im Bild ermittelt werden. In der Praxis wird dazu aber nicht nur ein konkreter Farbwert benutzt, sondern es wird mit Hilfe eines Schwellenwertes ein Farbraum zur Erkennung aufgespannt. Dieses Vorgehen ist erforderlich, weil in der Natur Objekte selten exakt einen einzigen Farbton besitzen. Statt dessen besitzen sie, auch in Abhängigkeit der Beleuchtung, verschiedene Nuancen einer Farbe. So ist rot nicht gleich rot, was anhand von Abbildung 3.1 leicht nachvollzogen werden kann.

Abbildung 3.1: Farbverlauf

[2] 'Computer Vision' – Bereich der Informatik, der sich mit der Erfassung und Verarbeitung optischer Informationen befasst. Salopp ausgedrückt ist das Ziel dieses Bereiches, dem Computer das 'Sehen' beizubringen.
[3] Pixel – Ein Kunstwort, das für "Picture Element" steht und die kleinste, sichtbare Einheit eines digitalen Bildes bezeichnet.

Für umfangreiche Informationen bezüglich der Zusammensetzung und Darstellung von Farben im Computer sei auf [Gierling-01] verwiesen und Beispiele für die Verfolgung ausgezeichneter Marker anhand von Farben sind unter anderem bei [Balzerowski-02, Smith-04] zu finden.

3.1.2 Geometrie des verfolgten Objektes

Objekte stellen ein weiteres Merkmal dar, das für die Erkennung und Verfolgung von Bewegungen herangezogen wird. Dabei kann es sich um einfache Objekte, wie z. B. einen Kreis, oder aber um komplexe wie das menschliche Gesicht handeln.
Um selbige nun innerhalb von Bildern lokalisieren zu können macht man sich ihre geometrischen Eigenschaften, wie Form und Positionierungen zueinander, zu nutze.

Das menschliche Gesichtes läßt sich beispielsweise durch seinen geometrischen Aufbau beschreiben. So hat es eine ovale Form, besitzt im Zentrum des unteren Drittels einen Mund, im Zentrum des mittleren Drittels eine Nase und an den Rändern des oberen Drittels die Augen. Eine weitere Eigenschaft ist, dass Mund und Nase im Zusammenhang mit den Augen jeweils ein Dreieck bilden.
Anhand dieser geometrischen Formen und Anordnungen kann ein Objekt schon grob als Gesicht klassifiziert werden.

Dieses Beispiel zeigt jedoch auch, dass das Erkennen eines komplexen Objektes (Gesicht) es erforderlich macht, gleich mehrere Objekte (Augen, Nase, Mund) erkennen zu müssen. Die Erkennung solcher Objekte führt dann, je nach ihrem Aufbau, zu einer entsprechend hohen Komplexität.

Für Beispiele der Erkennung anhand spezieller geometrischen Eigenschaften des verfolgten Objektes siehe [Turk-02, Gorodnichy-02]. Dort werden spezifische Merkmale des menschlichen Gesichtes verfolgt.

3.2 Schwierigkeiten der Erkennung

Aufgrund der mitunter hohen Komplexität, die die Verarbeitung und Auswertung optischer Informationen mit sich bringt, wird man bei der visuellen Gestenerkennung mit verschiedenen Schwierigkeiten konfrontiert. Daher sollen die nachfolgenden Punkte jeweils eine kurze Beschreibung der wesentlichen Problemfelder geben.

3.2.1 Hardware

Das grundlegendste Problemfeld bildet die Hardware, die zur Erfassung der Gesten verwendet wird. Von Ihr hängt nämlich direkt die Qualität des Ausgangsmaterials ab, das für die algorithmische Erkennung von Bewegungen - und somit von Gesten - zur Verfügung steht. Im Bereich der visuellen Erkennung handelt es sich dabei um die eingesetzten Kameras.

Bei ihrer Auswahl ist z. B. auf folgendes zu achten:

- sie muß die gewünschten Informationen (Infrarot-, Restlicht- oder Normalbilder) liefern können;
- sie muß eine möglichst hohe Auflösung bieten, damit auch feine Details zuverlässig erkannt werden können;
- sie muß die Bilder mit einer möglichst hohen Frequenz erfassen können um eine feine Granularität erfaßter Bewegungsabläufe zu erhalten.

Je genauer die genannten Anforderungen beachtet werden, um so besser wird auch das Ausgangsmaterial für die Bewegungs- und Gestenerkennung.

3.2.2 Beleuchtung

Die Beleuchtung der Umgebung wirkt sich durch Schatten und Reflexionen natürlich stark auf die grundlegende Fähigkeit zur Erkennung von Objekten aus. Besonders stark sind ihre Auswirkungen jedoch im Bezug auf das Erkennen von Farben bzw. Farbtönen. So kann das Erkennen von Objekten z. B. relativ leicht durch den zusätzlichen Einsatz von Infrarot-, oder Nachtsicht-Kameras verbessert werden.

Für die Farberkennung sind Beleuchtungseffekte jedoch regelrecht verheerend.
Starke Reflexionen von Licht auf spiegelnden Oberflächen sorgen dafür, dass die Farben der näheren Umgebung durch den auftretenden Blendeffekt stark verfälscht und sogar komplett überdeckt werden. Im Gegenzug sorgen Schatten dafür, dass die Farbtöne immer dunkler und dunkler erscheinen, bis sie schließlich in Abstufungen von grau und schwarz übergehen. Somit gehen gerade im Bereich der Farbtöne mitunter sehr viele Informationen verloren.

Ein weiteres Problem bereitet auch noch die Zusammensetzung des Lichts an sich. Natürliches und künstliches Licht besitzen unterschiedliche Blauanteile. Dadurch kann ein und dieselbe Farbe, in Abhängigkeit der Leuchtquelle, jeweils mit anderen Nuancen erkannt werden. Siehe dazu auch [Gierling-01] und "Kapitel 6.1" in [Balzerowski-02].

3.2.3 Perspektive

Eine weitere Schwierigkeit bei der Erfassung stellt die Perspektive der Kamera dar.
Die Komplexität des Erkennens der verfolgten Objekte und der Gesten hängt stark davon ab, wie weit die Objekte von der Kamera entfernt sind und welchen Winkel sie ihr gegenüber einnehmen. In Abhängigkeit dieser Faktoren treten z. B. folgende Phänomene auf:

- verfolgte Objekte werden überdeckt und sind somit nicht sichtbar;
- bedingt durch den Blickwinkel erscheinen Objekte perspektivisch verzerrt und sind daher schwerer zu lokalisieren;
- in Abhängigkeit von der Entfernung zur Kamera sind Bewegungen zur ihr hin bzw. von ihr weg nur schwer zu erkennen.

Um solchen Phänomenen entgegenzuwirken muß entweder der Erzeuger der Gesten dazu aufgefordert werden, die Gesten direkt und deutlich "in die Kamera" zu vollführen, oder es müssen mehrere Kameras eingesetzt werden um ihn aus verschiedenen Blickwinkeln zu beobachten. Letzteres ist dabei zu bevorzugen, da es zu besseren Ergebnissen führt und dem Benutzer mehr Freiheit einräumt. Allerdings erhöht sich dann auch der Aufwand, der für die Erkennung notwendig ist. Es müssen entsprechend mehr Bilder ausgewertet und die Kameras synchronisiert werden.

Das Problem der perspektivischen Verzerrung bleibt jedoch bei beiden Varianten bestehen und ist nur auf algorithmischem Wege zu beseitigen.

3.2.4 Feinheit der Gesten

Mit der Feinheit der Gesten ist gemeint, wie fein die Bewegungen sein dürfen, anhand derer eine Geste erkannt werden soll. Primär hängt dies davon ab, wie gut die Probleme der vorangegangenen Punkte gelöst werden, denn dadurch entscheidet sich, bis zu welchem Feinheitsgrad Bewegungen überhaupt erkannt werden. Deshalb bezieht sich dieser Punkt mehr auf Beachtung menschlicher Faktoren.

Für die Gestenerkennung sind auch Überlegungen anzustellen, was man den Benutzern überhaupt für Gesten ermöglichen bzw. abringen soll. Es ist z. B. schon ein gewisser Unterschied, ob das System Gesten erkennen kann, wenn der Benutzer seine Hand 10 cm bewegt, oder ob er sie 100 cm bewegen muß. Die Erkennung muß daher also möglichst feine Gesten akzeptieren. Allerdings brauchen selbige wiederum auch nicht zu fein zu sein, weil Menschen dazu neigen, unbewußt Bewegungen zu vollziehen. Diese Bewegungen äußern sich z. B. in einem leichten Zittern der Hände, oder einer Bewegung des Armes, usw. Somit entsteht gerade im Bereich feinster Gesten ein hohes Risiko zu Fehlinterpretationen.

3.2.5 Algorithmen

Im Rahmen dieses Punktes werden keine konkreten Algorithmen vorgestellt, sondern es soll nur darauf hingewiesen werden, dass die Auswahl bzw. Definition der Algorithmen eine weitere große Schwierigkeit bei der Bewegungs- und Gestenerkennung darstellt.

Ihre Auswahl, Arbeitsweise, sowie ihre Komplexität hängen zu stark von den vorangegangenen Punkten ab, als dass es "die" Standardlösung geben könnte. So ist z. B. die Objektlokation unter perspektivischen Verzerrungsbedingungen ohne Beachtung der Beleuchtung schon recht komplex.

Hinzu kommt, dass die Algorithmen im Bereich der Gestenerkennung Echtzeitanforderungen erfüllen müssen um eine akzeptable Ansprechzeit des Systems gewährleisten zu können.

Für konkrete Ansätze und Algorithmen sei auf entsprechende Fachliteratur verwiesen. Einen guten Ausgangspunkt stellt auch [ComputerVision-05] dar, wo zahlreiche Referenzen zur Bildverarbeitung zu finden sind.

4 Das System

In diesem Kapitel wird das eigentliche Konzept für die Bedienung eines PCs per Gestenerkennung ausgearbeitet.

4.1 Aufgabe

Es gilt ein System zu entwerfen, das in der Lage sein soll einen PC anhand von Gesten zu bedienen, welche der Benutzer vollführt. Um dies zu erreichen soll das System das PC-Betriebssystem bzw. dessen Oberfläche so instrumentieren, dass Anwendungen gestartet und / oder bedient werden.
Da die gängigen PC-Betriebssysteme im Home- und Office-Bereich heutzutage üblicherweise über eine grafische Benutzeroberfläche verfügen, soll die Bedienung der Anwendungen wahlweise per Emulation der Maus oder durch direkte Instrumentierung (API des Betriebssystems) stattfinden.

4.2 Analyse

In den nachfolgenden Unterpunkten werden zuerst einige grobe Rahmenbedingungen festgelegt unter denen die Software arbeiten können muß. Im Anschluß daran werden die Aufgabe und die Plazierung der Software anhand einer Black-Box-Ansicht dargestellt. Dann wird ein Marker für die Gestenerkennung vorgeschlagen und es werden spezielle Anforderungen an die Software spezifiziert.

4.2.1 Rahmenbedingungen

Im Rahmen dieses Punktes werden grundlegende Bedingungen festgelegt, welche von dem zu erstellenden System zu erfüllen sind. Sie sind allerdings noch recht grob gehalten, da diese Arbeit lediglich ein Konzept beschreibt und keine konkrete Implementierung für eine bestimmte Plattform. Zum Beispiel hängt die minimal benötigte Rechenleistung der Zielplattform stark von der gewählten Art der Bildverarbeitung zur Erkennung von Gesten ab.

4.2.1.1 Kosten

Um einen hohen Verbreitungsgrad und eine große Zielgruppe zu erreichen, sollte eine möglichst preisgünstige Lösung entstehen. Dies gilt es bei der Auswahl der Hardware und evtl. zu verwendender Softwarebibliotheken zu berücksichtigen.

4.2.1.2 Hardware

- Die Erfassung der Bewegungen und somit der Gesten soll über eine handelsübliche Webcam erfolgen, da diese sehr preiswert sind und recht gute Auflösungen liefern.
- Die benötigte Rechenleistung der Zielplattform soll einem handelsüblichen PC entsprechen, damit eine hoher Verbreitungsgrad gewährleistet werden kann.

4.2.1.3 Software

Das System sollte so weit wie möglich plattformunabhängig gehalten werden. Daher bieten sich interpretierte Sprachen wie z. B. die Programmiersprachen aus dem .Net-Framework™ [4] von Microsoft oder Java™ von Sun an.

4.2.2 Black-Box-Modell

Die nachfolgende Abbildung zeigt den grundsätzlichen Aufbau des zu erstellenden Systems im Rahmen einer Black-Box-Betrachtung.

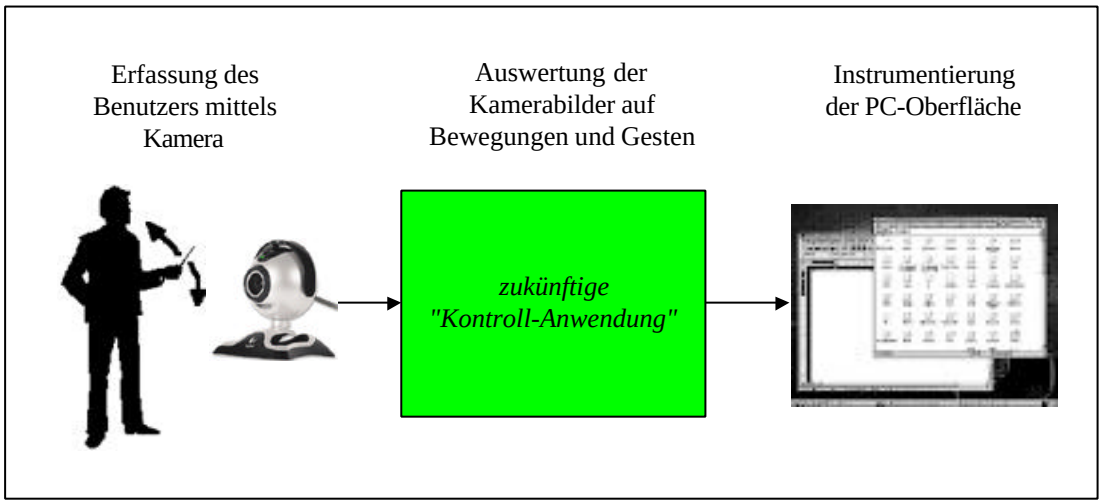

Abbildung 4.1: Grundsätzlicher Aufbau des Systems

Das System soll die Bewegungen des Benutzers mit Hilfe einer Kamera als Videostrom erfassen und anschließend auf Bewegungen und durchgeführte Gesten auswerten. Bei erkannten Gesten hat es das Betriebssystem bzw. dessen Oberfläche zu instrumentieren.

4.2.3 Vorschlag für Marker zur Gestenerkennung

Prinzipiell lassen sich Bewegungen und Gesten durch die Verfolgung eines beliebigen Objektes erkennen. Für ein möglichst intuitives System stellen jedoch die Hand bzw. die Hände die zu bevorzugende Variante dar, da der Mensch diese schon von Natur aus benutzt um zu gestikulieren.
Die Hände bieten zudem neben der reinen Gestenerkennung anhand ihrer Bewegungen als Ganzes noch weitere Möglichkeiten. Mit Hilfe der Finger lassen sich verschiedenste "Figuren" dastellen bzw. Posen einnehmen, so dass jede daraus resultierende Form der Hände jeweils einen Marker darstellen kann. Oder aber die Finger werden jeder für sich als weitere Marker betrachtet.

[4] Microsoft hat die Spezifikation der .Net-Laufzeitumgebung offengelegt, wodurch sie auch auf andere Plattformen portiert werden kann. Sie wird daher in naher Zukunft nicht nur auf den bekannten Betriebssystemen aus dem Hause Microsoft zur Verfügung stehen.

Die Menge der Marker bei der Hand als Ganzem und den Fingern als eigenständige Marker, kommt außerdem einer Maus-Emulation sehr entgegen. Man kann die Hand dann als die Maus und die Finger als deren Tasten betrachten.
Die Bewegung der Hand als Ganzes wäre der Bewegung des Zeigers auf dem Monitor gleichzusetzen. Die Maustasten wären bestimmten Fingern zuzuordnen (z. B. Zeigefinger linke Hand = linke Maustaste) und deren Betätigen erfolgt durch ausstrecken ("Drücken und gedrückt halten") und beugen ("loslassen") des jeweiligen Fingers.

Die Erkennung und Verfolgung dieser Vielzahl an Markern stellt jedoch auch relativ hohe Anforderungen an das System. Es muß über eine entsprechend hohe Auflösung seitens der Kamera verfügen und algorithmisch in der Lage sein, solch feine Details wie die Finger zuverlässig zu erkennen.
Um die Erkennung etwas zu vereinfachen könnte daher auch z. B. ein Handschuh herangezogen werden, bei dem die einzelnen Fingerkuppen unterschiedliche Farben haben. Die Farben würden dabei das Erkennen der Finger erleichtern.

4.2.4 Anforderungen an das System

In diesem Punkt werden die grundlegenden Anforderungen aufgelistet, die das System erfüllen muß. Als verwendeter Marker für die Erkennung ist dabei die Hand aus Punkt 4.2.3 mit Vereinfachung durch mehrfarbigem Handschuh zugrundegelegt worden.

- Das System muß über eine Benutzeroberfläche verfügen, die dessen Aktivierung, Deaktivierung und Konfiguration ermöglicht.
- Das System muß ausgezeichnete Marker innerhalb von digitalen Bildern einer Webcam anhand von Form und / oder Farbe finden können.
- Die Farben der Marker und dazugehörige Schwellenwerte müssen konfigurierbar sein, wobei sich deren Farbräume zwecks eindeutiger Identifizierung nicht überschneiden dürfen.
- Es muß die Bewegungen der Marker verfolgen können.
- Anhand der Markerbewegungen muß es Gesten erkennen können.
- Gesten müssen ein ausgezeichneten Anfang und ein Ende besitzen. Z. B. in der Form, dass für den Anfang ein bestimmter Marker sichtbar sein muß. Das Ende wird angenommen, sobald dieser nicht mehr sichtbar ist.
- Gesten müssen konfigurierbar sein. Dazu müssen bestehende Gesten gelöscht und neue definiert werden können.
- Neue Gesten sind dem System per Vorzeichnen durch den Benutzer beizubringen, indem sie vom Benutzer vor der Kamera durchgeführt und vom System aufgenommen werden. Bei der Aufnahme soll zur Kontrolle eine "Vorschauanzeige" der erkannten Geste für den Benutzer dargestellt werden.
- Es muß eine Toleranzschwelle einstellbar sein, bis zu der eine vom Benutzer durchgeführte und eine dem System bekannte Geste als übereinstimmend anerkannt wird.
- Das System sollte von Haus aus schon einen Satz von Standardgesten mitbringen (siehe 4.3.4).

- Das System muß Instrumentierungsaktionen bereitstellen, mit denen die Oberfläche der Zielplattform instrumentiert werden kann. Dazu gehören Aktionen zum Emulieren von Maus- und Tastaturereignissen, Aktionen zum Starten von Programmen und zum direkten Instrumentieren der Fenster. Siehe 4.2.5.
- Eine oder mehrere Instrumentierungsaktionen müssen zu einem Kommando zusammengefaßt werden können.
- Ein Kommando soll aus Instrumentierungsaktionen und Kommandos bestehen können. (Composite-Pattern)
- Ein Kommando muß einer oder mehreren Gesten zugeordnet werden können.
- Einer Geste muß entweder ein oder kein Kommando zugeordnet sein.
- Das System muß über vier Betriebsmodi verfügen:
  1 – inaktiv
  2 – aktiv als reine Gestenerkennung zur Ausführung von Kommandos (Kommando-Modus)
  3 – aktiv als Maus-Emulation (Maus-Modus)
  4 – aktiv zur Gestenaufzeichnung (Aufzeichnungs-Modus)
- Im Kommando-Modus muß das System bei einer erkannten Geste dessen zugeordnetes Kommando (sofern vorhanden) ausführen. (Command-Pattern)
- Im Maus-Modus ist anhand der Marker keine Erkennung von Gesten wie im Kommando-Modus durchzuführen. Statt dessen erfolgt eine kontinuierliche Verfolgung mit entsprechender Umsetzung in eine Bewegung des Mauszeigers auf der Oberfläche der Zielplattform.
- Für den Maus-Modus müssen einzelne Marker bzw. Markerkombinationen den Eigenschaften der Maus zugeordnet werden können.
  – Marker für Bewegungserkennung
  – linke Maustaste
  – mittlere Maustaste
  – rechte Maustaste
  – Mausrad
- Die Empfindlichkeit der "visuellen Maus" muß konfigurierbar sein. Das heißt, es muß einstellbar sein, wie empfindlich das System die Bewegungen des Benutzers in Bewegungen des Mauszeigers umsetzt.
- Das System muß zwischen den Modi mittels User-Interface des Systems wechseln können.
- Das System muß zwischen den Modi Maus-Modus und Kommando-Modus auch mittels visueller Eingaben umschaltbar sein. Dazu muß ein Marker bzw. eine Markerkombination definierbar sein.
- Im Kommando-Modus hat das Erfassen von Gesten zu beginnen, sobald irgendein Marker sichtbar ist. Davon ausgenommen ist die spezielle Markerkombinationen für den Moduswechsel.
- Sämtliche Konfigurationen müssen persistent gehalten werden.

4.2.5 Aktionen zur Instrumentierung der Zielplattform

Um eine umfangreiche Instrumentierung der Oberfläche der Zielplattform zu erhalten muß das System parametrisierbare Aktionen bereitstellen. Die Aktionen müssen es ermöglichen, Anwendungen zu starten, Fenster direkt zu manipulieren bzw. zu bedienen, sowie Tastatur- und Mausereignisse zu emulieren und an Anwendungen zu senden.

Der Umfang der bereitzustellenden Aktionen und deren Parametrisierung hängt stark von den Möglichkeiten ab, welche die API der Zielplattform überhaupt zur Umsetzung erlaubt. Daher sind die nachfolgenden Aktionen recht allgemein gehalten.

Tastatur

- Taste gedrückt mit Übergabe der Taste - z. B. "KeyDown(e)"
- Taste losgelassen mit Übergabe der Taste - z. B. "KeyUp(e)"
- Tastendruck (Kombination von gedrückt und losgelassen) - z. B. "KeyPress(e)"

Maus

- Mausbewegung mit Übergabe der Richtung
- Taste gedrückt
- Taste losgelassen
- Tastendruck (Kombination von gedrückt und losgelassen)
- Tasten-Doppelklick
- Mausradbewegung mit Übergabe der Richtung

(Anwendungs-)Fensterbedienung

- Fenster finden anhand von Fenstertitel oder anderem Merkmal
- Fenster maximieren
- Fenster minimieren
- Fenster schließen
- Fenster aktivieren / deaktivieren (Fokuswechsel)

Allgemein

- Starten von Programmen mit Übergabe von Dateinamen und Parametern

4.3 Design

Im Rahmen dieses Punktes wird ein mögliches Design für das System entworfen.
Es werden feinere Modelle gezeigt, welche die Arbeitsweise und den internen Aufbau des Systems darstellen. Anschließend werden die einzelnen Module beschrieben und die wichtigsten Zustands- und Ablaufdiagramme vorgestellt. Zum Schluß erfolgt noch die Auswahl der Gesten für das System.

4.3.1 Feinere Modellierung

In diesem Punkt wird das System und dessen interner Aufbau anhand von Modellen dargestellt.

Abbildung 4.2 zeigt das Zusammenspiel zwischen dem Instrumentierungssystem und seiner Umgebung in Form einer feineren Black-Box-Betrachtung.

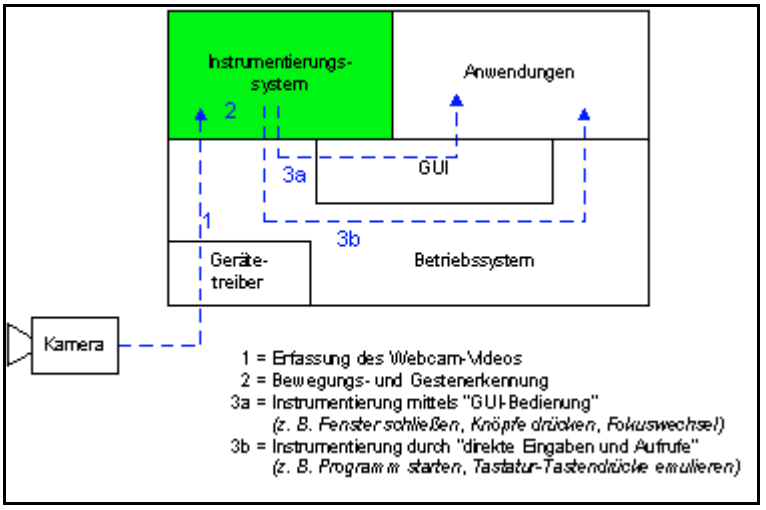

Abbildung 4.2: Grundlegendes Zusammenspiel mit der Zielplattform

Sie stellt den allgemeinen Informationsfluß und die unterschiedlichen Arten der Instrumentierung dar, mit denen Anwendungen gesteuert werden können. Die Steuerung erfolgt über Methoden, die das API des Betriebssystems zur Verfügung stellt um Ereignisse der Maus, Tastatur und der grafischen Oberfläche zu erzeugen bzw. um sie dem System bekannt zu machen. Über diesen Weg werden sie dann an die zu steuernde Anwendung gesendet.

In Abbildung 4.3 erfolgt der Übergang von der Black-Box-Ansicht zu einer genaueren Ansicht vom internen Aufbau des Instrumentierungssystems. Sie stellt die einzelnen Module dar, aus denen die Software bestehen soll.

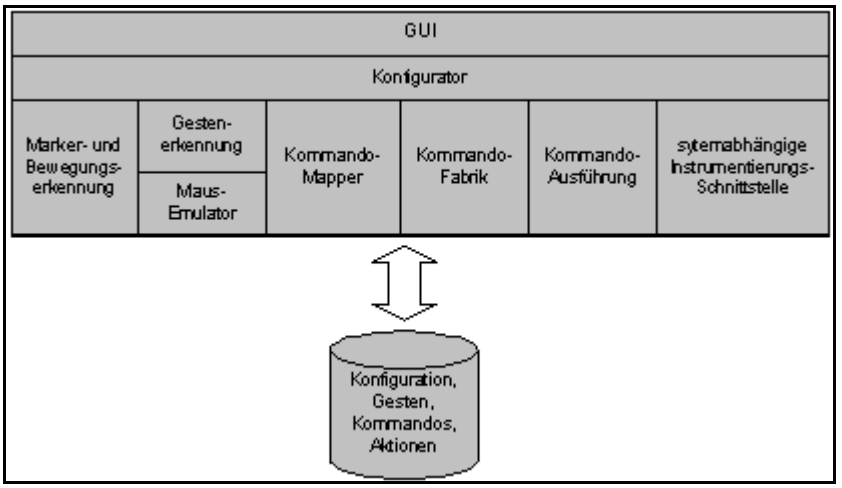

Abbildung 4.3: Die Module des Systems

4.3.2 Die Module

GUI
Dieses Modul stellt, wie bei den meisten Anwendungen, das Interface zwischen dem Benutzer und dem System dar. Mit seiner Hilfe soll der Benutzer die Bedienung, Aktivierung / Deaktivierung und Konfiguration des Systems vornehmen.

Konfigurator
Der Konfigurator soll die Konfiguration des Systems kapseln. Für das Laden und Speichern bestimmter Optionen müssen sich alle anderen Module an den Konfigurator wenden. Die Module müssen sich bei ihm registrieren und werden von ihm über Änderungen, die der Benutzer an der Konfiguration vornimmt, informiert.

Marker- und Bewegungserkennung
Dieses Modul ist für den Empfang der Kamerabilder und deren Verarbeitung zuständig. Es muß mittels grafischer Verfahren die Marker und ihre Bewegungen erkennen.
Erkannte Marker und Bewegungen hat es als Paare der Form Marker-Bewegung, je nach aktuellem Betriebsmodus, an die Gestenerkennung und den Maus-Emulator zu übergeben.

Gestenerkennung
Die Gestenerkennung erstellt eine Historie über die Marker-Bewegung-Paare, die ihr übergeben werden, und wertet diese auf das Vorhandensein von Gesten aus. Wenn eine Geste gefunden wurde reicht sie eine ID, anhand derer die Geste identifiziert werden kann, an den Kommando-Mapper weiter.
Im Modus der Gestenaufzeichnung ist die Gestenerkennung für das Aufzeichnen der durchgeführten Geste zuständig.

Maus-Emulator
Der Maus-Emulator wertet die übergebenen Marker-Bewegung-Paare aus und führt, in Abhängigkeit der jeweiligen Marker (Bewegung, Tasten etc.), entsprechende Instrumentierungsaktionen mit Hilfe der systemabhängigen Instrumentierungs-Schnittstelle aus.
Für die Bewegung des Mauszeigers ermittelt er dessen neue Position, indem er ein Mapping der erkannten Bewegung mit der aktuellen Position des Mauszeigers vornimmt und selbige entsprechend anpaßt.
Das direkte Instrumentieren aus dem Modul heraus wurde gewählt, damit die "visuelle Maus" möglichst ohne große Verzögerung auf die Bewegungen des Benutzers reagieren kann.

Kommando-Mapper
Der Kommando-Mapper setzt Gesten-IDs in IDs von Kommandos um, welche für die jeweiligen Gesten hinterlegt wurden. Die Kommando-IDs reicht er an die Kommando-Fabrik weiter.

Kommando-Fabrik
Die Fabrik erstellt für die eingehenden Kommando-IDs die zugehörigen Kommando-Objekte und reicht sie an die Kommando-Ausführung weiter.

Kommando-Ausführung
In der Kommando-Ausführung werden die Kommando-Objekte abgearbeitet. Dazu werden die Aktionen des Kommando-Objektes mit Hilfe der systemabhängigen Instrumentierungs-Schnittstelle ausgeführt.

Systemabhängige Instrumentierungs-Schnittstelle
Diese Schnittstelle führt die Aktionen für die Instrumentierung aus. Dazu setzt es die Aktionen in Aufrufe konkreter Methoden der Zielplattform um.

4.3.3 Diagramme

4.3.3.1 Diagramm: Zustandswechsel der verschiedenen Betriebsmodi

Dieses Diagramm zeigt die Übergänge zwischen den vier Betriebsmodi Inaktiv, Aufnahme-Modus, Kommando-Modus und Maus-Modus.

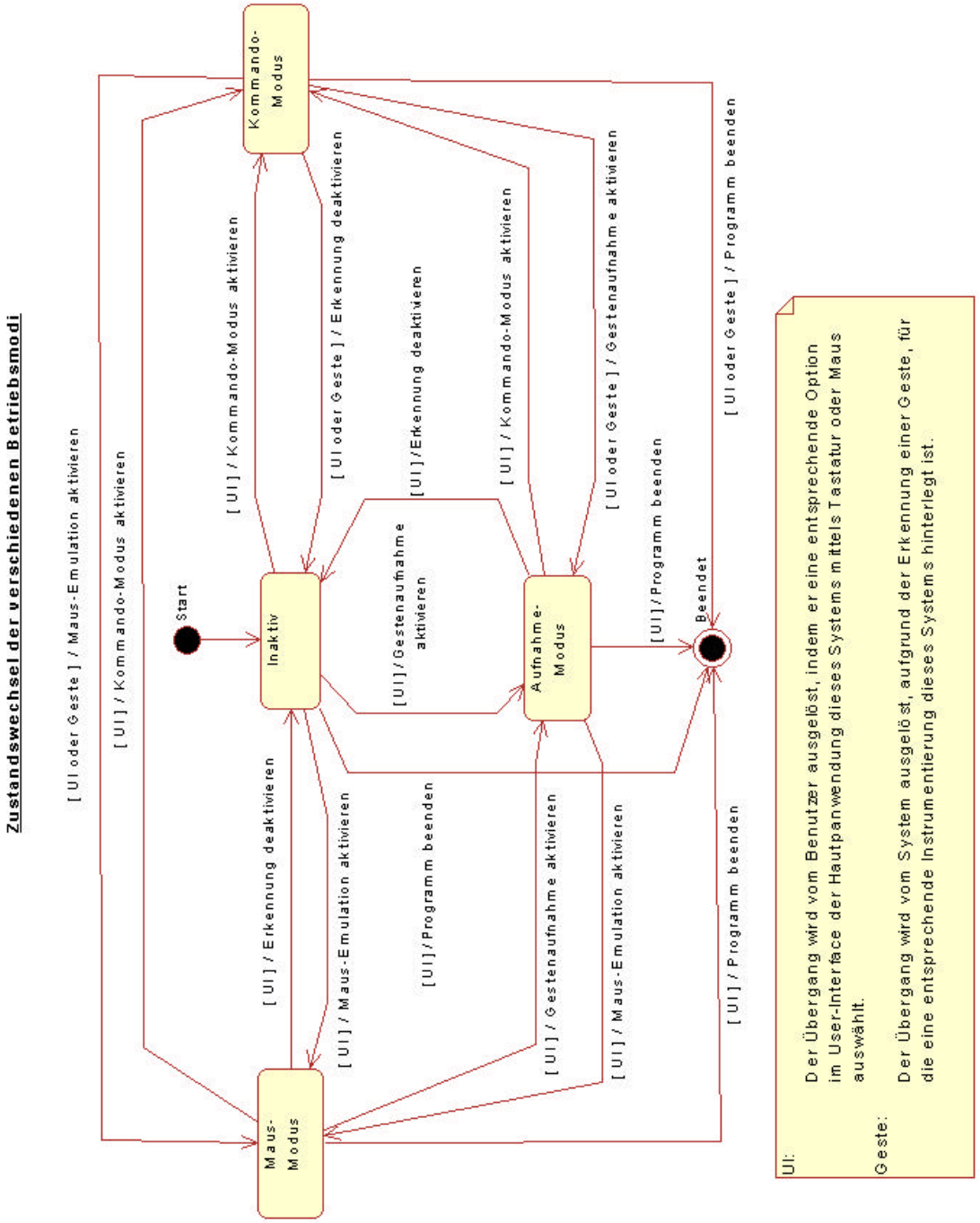

Abbildung 4.4: Zustandswechsel der vier Betriebsmodi

4.3.3.2 Ablaufdiagramm: Kommando-Modus

Dieses Diagramm zeigt den Ablauf bei Erkennung einer Geste.

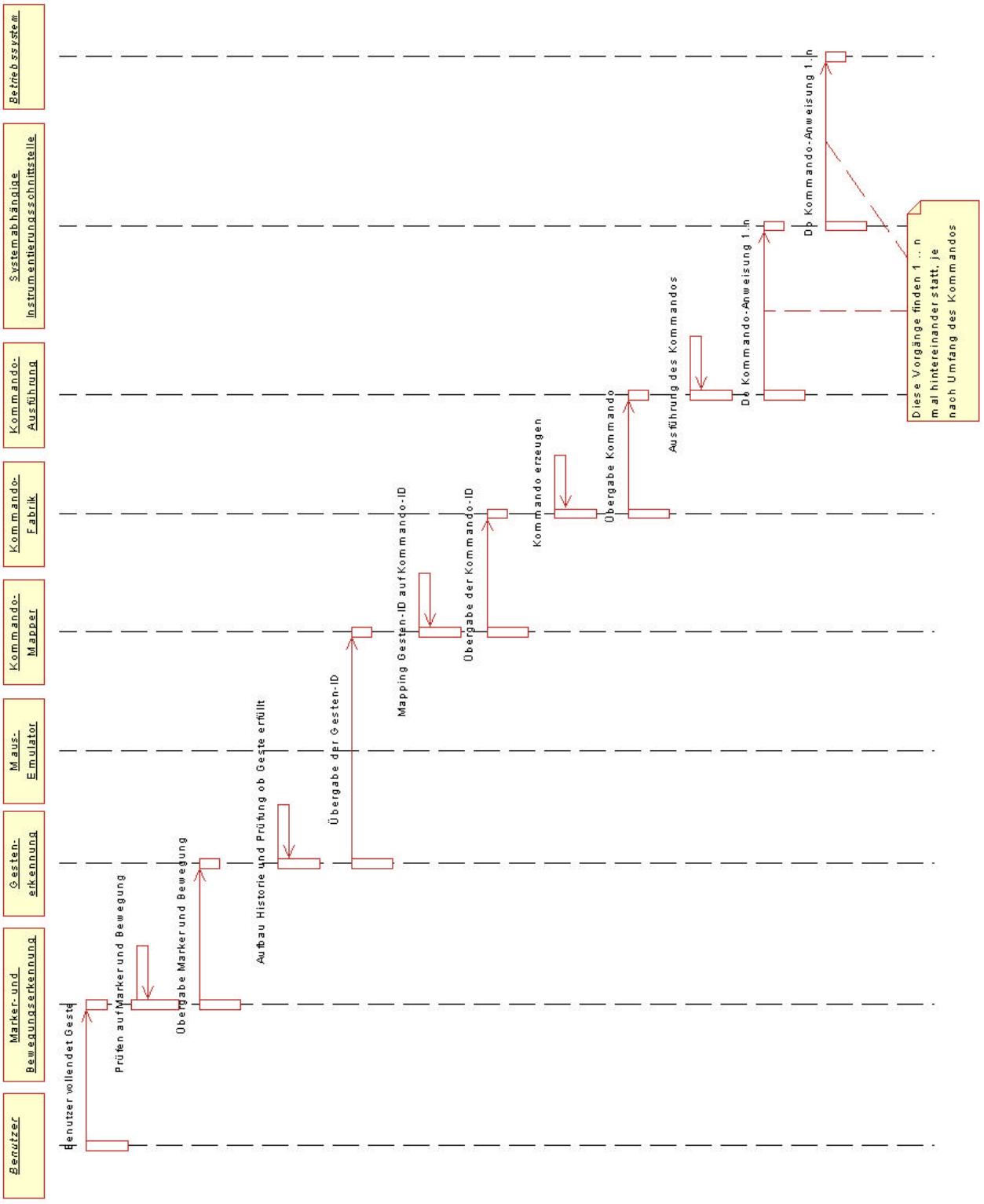

Abbildung 4.5: Ablaufdiagramm des Kommando-Modus

4.3.3.3 Ablaufdiagramm: Maus-Modus

Dieses Diagramm zeigt den Ablauf bei Erkennung einer Bewegung der "Maus".

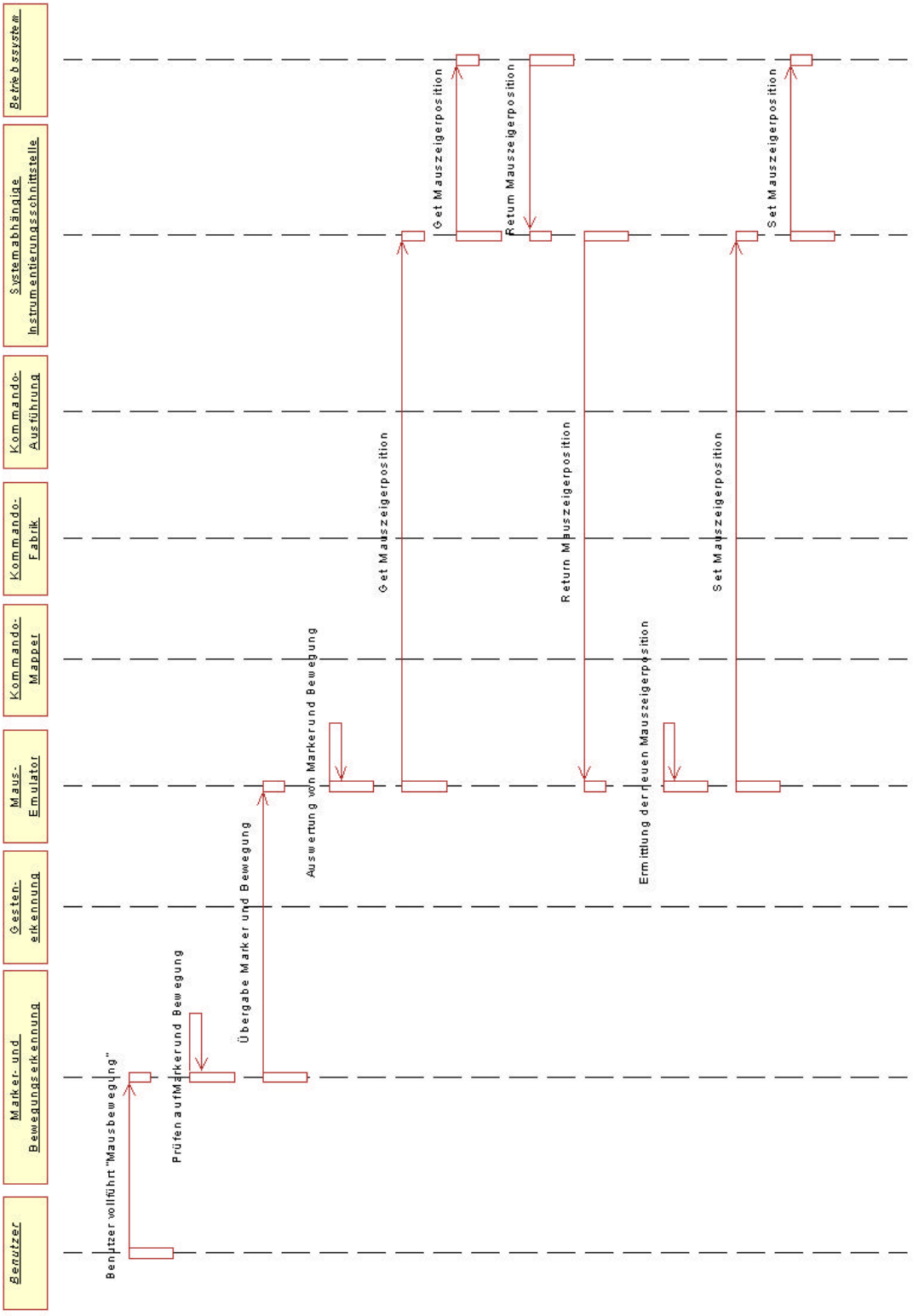

Abbildung 4.6: Ablaufdiagramm des Maus-Modus

4.3.4 Auswahl der Gesten

An dieser Stelle muß leider gesagt werden, dass es ohne ein konkretes Einsatzgebiet nicht möglich ist bestimmte Gesten zu definieren. Sie hängen zu stark von mehreren Faktoren ab.
Da sind zum Beispiel die zu instrumentierenden Anwendungen, in deren Kontext die Gesten einen Sinn für den Benutzer ergeben müssen um den Lernaufwand gering zu halten. Hinzu kommen auch Präferenzen und körperliche Möglichkeiten der Benutzer. Z. B. kann nicht jeder seine Finger wirklich einzeln bewegen und somit jeden Finger beliebig strecken und beugen. Der eine möchte vielleicht seinen Webbrowser mit einer geraden Bewegung von links nach rechts starten, der andere mit einer kreisförmigen Bewegung von rechts nach links.

Mit den Gesten verhält es sich wie mit der Sprache. So gibt es Mehrdeutigkeiten in der Form, dass Menschen unterschiedliche Gesten machen um ein und dieselbe Sache auszudrücken. Das hängt unter anderem stark mit dem Umfeld zusammen aus dem sie jeweils kommen.

Ein weiterer Punkt sind kulturelle Unterschiede bzw. Gepflogenheiten. So sind je nach Umfeld und Land bestimmte Gesten aus politischen oder pietätbedingten Gründen zu vermeiden.

Es erscheint daher am sinnvollsten, die Gesten konfigurierbar zu halten und von Haus aus nur einen kleinen Standardsatz wie in Abbildung 4.2 anzubieten. Zum Einen können dann für ein Verbreiten des Systems leicht lokalisierte Gestensätze erstellt werden und es bietet den Benutzern die komfortable Möglichkeit, die Gesten ihrem Geschmack entsprechend zu gestalten.

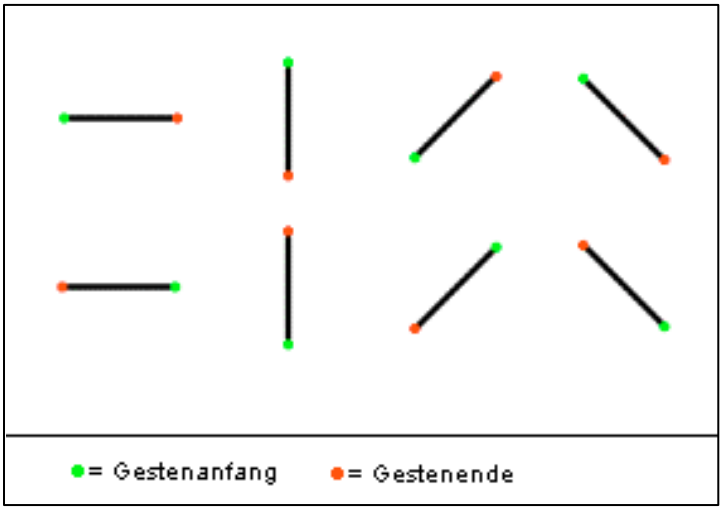

Abbildung 4.7: Mögliche Standardgesten

5 Ausblick

Das spezifizierte System dieser Arbeit ermöglicht die teilweise Bedienung eines Computers mittels Gesten. So erlaubt es zum Einen die Bedienung der Oberfläche im Sinne eines vollwertigen Ersatzes für die Maus, und zum Anderen können mit Hilfe von Gesten bestimmte dedizierte Geräte sogar vollständig bedient werden.
Zur vollständigen Bedienung eines allgemeinen Computers ist es leider nicht in der Lage, da Texteingaben nur schwer und umständlich mittels Gesten umzusetzen sind.

Für die Umsetzung von Texteingaben wäre eine Kopplung des Systems mit einer Spracherkennung wünschenswert. Das würde dann die Bedienung vollständig von Maus und Tastatur lösen und ein fast vollständig "mobiles Interface" ermöglichen.

Eine Notlösung für Texterfassung könnte aber auch ein Touchscreen darstellen. Notlösung deshalb, weil dadurch die Bedienung durch Gesten teilweise relativiert wird.

Für eifrige Tüftler könnte aber auch der nachfolgende Ansatz interessant sein...
Um Text zu erfassen wird ein Tastenfeld auf einen beliebigen Gegenstand projiziert. Der Benutzer "drückt" dessen Tasten indem er darauf zeigt. Das System erfaßt dies und setzt die Texteingabe um.
Wäre doch nett, wenn mobile Geräte dazu in der Lage wären, oder?

Das Feld der alternativen Eingabemöglichkeiten für die "Disappearing Computer" bietet noch allerlei Spielraum für Innovationen. Daher können wir gespannt sein, was für Varianten innerhalb der nächsten Jahre das Licht der Welt erblicken werden. ☺

6 Abbildungsverzeichnis

7 Literaturverzeichnis

[Balzerowski-02]
Balzerowski, R.: "Realisierung eines Webcam basierten Kamera-Systems für mobile Roboter", 2002.
http://users.informatik.haw-hamburg.de/~lego/Projekte/Balzerowski/diplomarbeit-www.pdf
Zugriffsdatum: 11.05.2005

[ComputerVision-05]
"The computer vision homepage".
http://www-2.cs.cmu.edu/~cil/vision.html
Zugriffsdatum: 13.05.2005

[DisappearingComputer-05]
"The Disappearing Computer Initiative".
http://www.disappearing-computer.net
Zugriffsdatum: 08.05.2005

[Gierling-01]
Gierling, W.: "Farbmanagement", Bonn: mitp-Verlag, 2001.
ISBN 3-8266-0679-5

[Gorodnichy-02]
Gorodnichy, D. O. und Roth, G.: "Affordable yet robust and precise face tracking using USB cameras with application to designing handsfree user interfaces", Computational Video Group, IIT, NRC, 2002
http://www.cv.iit.nrc.ca/research/Nouse/docs/uist.pdf
Zugriffsdatum: 08.05.2005

[Nielsen-93]
Nielsen, J.: "Noncommand user interfaces", Communications of the ACM, 36(4):83-99, April 1993.
http://www.useit.com/papers/noncommand.html
Zugriffsdatum: 12.05.2005

[Russel-05]
Russell, D. M., Streitz, N. A. und Winograd, T.: "Building Disappering Computers", Communications of the ACM, 48(3):42-48, März 2005.

[Siebel-03]
Siebel, N. T.: "Design and Implementation of People Tracking Algorithms for Visual Surveillance Applications", 2003
http://www.siebel-research.de/publications/Siebel-thesis-onesided.pdf
Zugriffsdatum: 07.05.2005

[Siebel-05]
Siebel, N. T.: "The Reading People Tracker", 2005
http://www.siebel-research.de/people_tracking/reading_people_tracker/
Zugriffsdatum: 07.05.2005

[Smith-04]
Smith, R., Piekarski, W. und Wigley, G.: "Hand Tracking For Low Powered Mobile AR User Interfaces", Wearable Computer Lab / Reconfigurable Computing Laboratory, School of Computer and Information Science, University of South Australia, 2004
http://www.tinmith.net/papers/smith-auic-2005.pdf
Zugriffsdatum: 08.05.2005

[Streiz-05]
Streitz, N. und Nixon, P.: "Disappering Computer", Communications of the ACM, 48(3):32-35, März 2005.

[Turk-02]
Turk, M. et. al.: "TLA Based Face Tracking", Computer Science Department - Psychology Department, University of California, 2002
http://ilab.cs.ucsb.edu/projects/turk/Turk%20et%20al%202002.pdf
Zugriffsdatum: 08.05.2005

[Weiser-91]
Weiser, M.: "The Computer for the 21st Century", Scientific American, S. 66–75, September 1991.