Otto-von-Guericke-Universität Magdeburg

Fakultät für Informatik

Masterarbeit

Erkennung von formgleichen Gesten mit unterschiedlichen zeitlichen Dynamiken mittels Dynamic Time Warping

Autor:

Patrick Haese

19. März 2017

Betreuer:

Dr. Claudia Krull

Institut für Simulation und Graphik

Prof. Dr. Rudolf Kruse

Institut für Intelligente Kooperierende Systeme

Haese, Patrick:
Erkennung von formgleichen Gesten mit unterschiedlichen zeitlichen Dynamiken mittels Dynamic Time Warping
Masterarbeit, Otto-von-Guericke-Universität Magdeburg, 2017.

Inhaltsangabe

untersucht ob formgleiche Gesten mittels Dynamic Time Warping anhand ihrer zeitlichen Dynamiken unterschieden werden können. Experimente deuten darauf hin, dass das Erkennen von zeitlichen Dynamiken mittels DTW zwar theoretisch möglich ist, jedoch aktuell nur unzuverlässige Resultate liefert. Damit wäre es für reale Anwendungen nicht geeignet.

Inhaltsverzeichnis

1. Einführung

1.1 Hintergrund

In den vergangenen Jahren ist die automatische Erkennung von menschlichen Gesten zu einem wichtigen Bestandteil im Bereich der Human-Computer-Interaction geworden. Gesten gelten als bedeutungsvolle Bewegungen, die dazu dienen Informationen auszudrücken oder mit der Umgebung zu interagieren [MA07]. Nicht umsonst werden sie daher als wichtiger Zukunftsbaustein gesehen, um die Interaktion mit Computern einfacher und natürlicher zu gestalten.

Bisher verfügbare Systeme zur Gestenerkennung bedienen sich in der Hardware-Frage meist eines Handschuh- oder eines Kamera-basierten Ansatzes. *[Bilder]* Spezielle Handschuhe, die während des Ausführens von Gesten getragen werden müssen, sog. Data Gloves [PMS+09], enthalten Sensoren, die die Aufzeichnung selbst von komplexen Bewegungen ermöglichen. Im Gegensatz dazu stehen Systeme, wie z.B. Microsofts Kinect, die Kameras nutzen um Bewegungen aufzuzeichnen und die Position von Körperteilen im 3D-Raum zu verfolgen [LJ09].

Im Bereich der Software gibt es aktuell eine ganze Reihe von verschiedenen Methoden und Algorithmen, die erfolgreich zur Erkennung von Gesten angewandt werden. Zwei bekannte Vertreter sind zum Beispiel Hidden Markov Models (HMMs) [NKW96] oder das Dynamic Time Warping (DTW) [tHRH07].
HMM ist ein stochastisches Modell, bei dem auf Grundlage von externen Beobachtungen, Rückschlüsse auf interne Zustandsänderungen eines Systems gezogen werden. Beim DTW handelt es sich dagegen um einen Pattern-Matching-Ansatz, der genutzt wird um die Ähnlichkeit von zwei zeitabhängigen Signalfolgen mit unterschiedlicher Länge zu bestimmen [MÖ7].

Zeigt vielversprechende Ergebnisse, z.B. konnten bereits die verschiedenen Formen einer ganzen Zeichensprache erkannt werden [BCD06].

1.2 Motivation

Für manche Anwendungen kann es sinnvoll sein, Gesten nicht nur anhand ihrer Form zu erkennen, sondern auch verschiedene Muster in der Ausführungsgeschwindigkeit unterscheiden zu können. Ein mögliches Szenario wäre zum Beispiel eine Nutzung der Ausführungsgeschwindigkeit als zusätzliches Sicherheitsmerkmal bei gestenbasierten Verifikationsverfahren. Ebenfalls denkbar wäre ein Nutzen für eBooks, wobei eine höhere Ausführungsgeschwindigkeit von Wisch-Gesten dafür sorgt, dass mehre Seiten auf einmal umgeblättert werden können.

Bisher lag der Fokus hauptsächlich auf Hidden non-Markovian Models [BKH11] und Conversive hidden non-markovian Models [DKH15], zwei Weiterentwicklungen von HMMs, um Ausführungsgeschwindigkeiten von formgleichen Gesten zu unterscheiden. Aktuelle Forschungen konnten nun zeigen, dass auch HMMs in der Lage sind, zeitliche Dynamiken von formgleichen Gesten zu unterscheiden [Mar17].

Während es mit HMM-basierten Verfahren bereits mehrere Möglichkeiten gibt, zeitlich Dynamiken von Gesten zu unterscheiden, ist nicht klar ob andere Verfahren dazu ebenfalls in der Lage sind. Daher steht in dieser Arbeit auch die Frage im Zentrum, ob und, wenn ja, wie gut DTW in der Lage ist, Gesten anhand ihrer Ausführungsgeschwindigkeiten zu unterscheiden.

1.3 Ziele und Forschungsfragen

Aus den in Abschnitt 1.2 genannten Gründen ist es das Hauptziel dieser Arbeit zu untersuchen, ob formgleiche Gesten unter Nutzung von DTW anhand ihrer zeitlichen Dynamik unterschieden werden können.

Eine weitere offene Frage die im Rahmen dieser Arbeit untersucht werden soll, ist, ob zusammen mit einer Geste auch die ausführende Person identifiziert werden kann. Dies kann ebenfalls für gestenbasierten Verifikationsverfahren von Bedeutung sein.

Auch ist nicht klar in wie fern DTW geeignet ist um Gesten auf der Grundlage von fehlerhaften bzw. unvollständigen Daten zu erkennen. In der Realität kann es aufgrund technischer Limitierungen z.B. zu Aussetzern oder Verzögerungen bei der Aufzeichnung von Gesten kommen, was eine nachfolgende Erkennung erschweren oder gar verhindern könnte. Daher soll untersucht werden, ob die Erkennung von Gesten mit unvollständigen Daten negativ beeinflusst wird.

1.4 Aufgaben und Rahmenbedingungen

Um die in Abschnitt 1.3 formulierten Ziele zu erreichen sind mehrere Untersuchungen notwendig. Zunächst muss ein geeigneter auf DTW basierender Algorithmus zur Gestenerkennung ausgewählt werden. Ebenfalls müssen verschiedene Ansätze zur Simulation fehlerhafter Daten bzw. zur Modellierung zeitlicher Dynamiken entwickelt werden.

Um für spätere Experimente die Vergleichbarkeit zu gewährleisten, müssen Daten erstellt und aufgezeichnet werden. Zunächst muss ein Satz von verschiedenen Testgesten festgelegt werden. Der Gestensatz muss so beschaffen sein, das er sowohl

leicht als auch schwer unterscheidbare Gestenformen, als auch Gesten mit der gleichen Form aber unterschiedlichen Mustern in der Ausführungsgeschwindigkeit enthält. Um eine ausreichende Menge an Referenzdaten zu erlangen, werden die Gesten mehrfach ausgeführt uns idealerweise von mehreren Personen aufgezeichnet, um eine realistischere Diversität zu erreichen. Die Daten wurden sie mit Hilfe von Microsofts Kinect, also einem Kamera basierten System, aufgezeichnet und für die spätere Verwendung, in Form von verschiedenen Experimenten, in einer Datenbank gespeichert.

Zum Schluss werden die aufgezeichneten Daten mit dem Algorithmus in Kapitel 4 getestet um die zuvor in Abschnitt 1.3 gestellte Frage zu beantworten.

Der Rest der Arbeit gliedert sich wie folgt: In Kapitel 2 wird unter anderem kurz die Arbeitsweise von DTW erläutert und ein Ausblick gegeben, welche Methoden neben DTW im Feld der Gestenerkennung angewandt wurden. Kapitel 3 beinhaltet Erklärungen zum DTW-Algorithmus, der in dieser Arbeit verwendet wurde und zum Gestensatz, der als Datenquelle für die Experimente genutzt wurde. In Kapitel 4 werden dann alle Experimente, die zur Beantwortungen der formulierten Ziele durchgeführt wurden, beschrieben und deren Resultate erläutert. Zum Schluss werden in Kapitel 5 alle Ergebnisse zusammengefasst und bewertet in wie fern die Ziele erreicht wurden.

2. Grundlagen

In diesem Kapitel werden notwendige Grundlagen erklärt. Zuerst wird eine kurze Erklärung zu Kinect gegeben, dem Tool, das im Rahmen dieser Arbeit genutzt wurde, um Gesten aufzuzeichnen. Darauf folgt eine ausführliche Erklärung worum es sich bei DTW handelt und nach welchen Prinzipien es arbeitet. Zum Schluss wird beschrieben wie DTW im Feld der Gestenerkennung zuvor angewandt wurde und welche anderen Ansätze zur Gestenerkennung bisher genutzt wurden.

2.1 Kinect

Im Rahmen dieser Arbeit wurde Microsofts Kinect genutzt, um Gesten aufzuzeichnen. Mithilfe eines auf Infrarot-Lasern basierenden Tiefensensors ist Kinect in der Lage den gesamten menschlichen Körper zu erfassen und auf ein virtuelles 3D-Skelett abzubilden, wie in Abbildung 2.1 dargestellt. Die Position im dreidimensionalen Raum kann von jedem Punkt dieses Skeletts bestimmt werden.

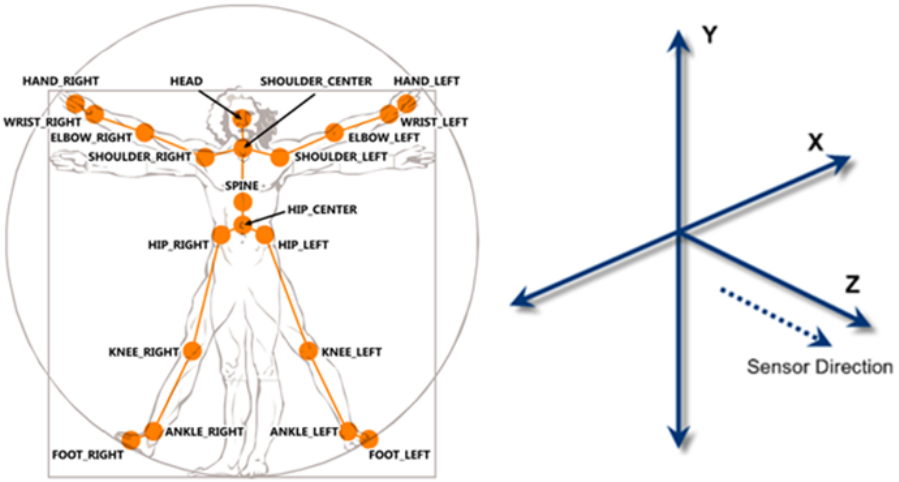

Abbildung 2.1: Repräsentation des menschlichen Skeletts mit Kinect und Koordinatensystems des 3D Raumes [KSa]

Die in dieser Arbeit genutzten Gesten wurden mit der rechten Hand der Versuchspersonen aufgezeichnet. Eine aufgezeichnete Geste ist dann eine Folge von dreidimensionalen Koordinaten der rechten Hand. Kinect ist in der Lage Bilder mit einer Frequenz von 30 Hertz aufzuzeichnen [SFC+11]. Folglich bilden dann 30 Koordinaten die Bewegung innerhalb einer Sekunde ab.

2.2 Dynamic Time Warping

2.2.1 Vergleich von Zeitreihen

Dynamic Time Warping ist ein Verfahren, mit dem die Ähnlichkeit von zwei Zeitreihen unterschiedlicher Länge zu bestimmt werden kann [MÖ07]. Beim Warping, also dem Verzerren einer Zeitreihe, werden, wie in Abbildung 2.2 dargestellt, die Punkte von einer Testsequenz auf die Punkte einer Referenzsequenz abgebildet, die den geringsten Abstand zueinander haben.

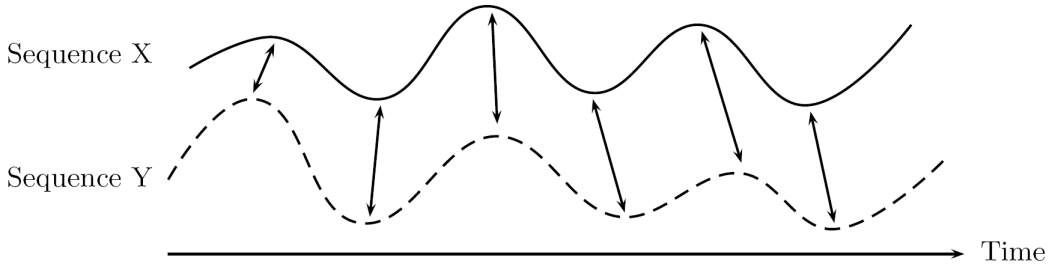

Abbildung 2.2: Darstellung eines Vergleichs zweier Seqzenzen mittels DTW [MÖ07]

Gegeben seien zwei Zeitreihen Q und R, der Länge n und m, deren Ähnlichkeit zu bestimmen ist.

$$Q = q_1, q_2, ..., q_i, ..., q_n$$
$$R = r_1, r_2, ..., r_j, ..., r_m$$

Als Maß für die Ähnlichkeit der zwei Zeitreihen dient die sogenannte warping-Distanz, die ausdrückt wie stark Q verzerrt werden muss, um optimal auf R abgebildet zu werden. Q kann dabei auf exponentiell viele Arten verzerrt werden, von denen jedoch die meisten ineffizient sind. Um den optimalen Warp zu finden, wird daher mit den Prinzipien der dynamischen Programmierung die $n \times m$ Abstandsmatrix DTW iterativ nach Formel 2.1 berechnet.

$$DTW[i,j] := d(q_i, r_j) + min\{DTW[i, j-1], DTW[i-1, j], DTW[i-1, j-1]\} \quad (2.1)$$

Bei d handelt es sich dabei um ein zuvor gewähltes Abstandsmaß, mit dem der Abstand zwischen zwei Punkten von Q und R berechnet wird. Die Kosten für den nächsten Warping-Schritt setzen sich also zusammen aus den Kosten für den bisher günstigsten Teilpfad und dem Abstand zwischen den zwei aufeinander abzubildenden Punkten. Nachdem die Abstandsmatrix DTW vollständig berechnet wurde, kann dann der optimale Warping Pfad mittels Backtracking ermittelt werden.

2.2.2 Klassifikation mit DTW

Für eine reine Anwendung im Feld der Klassifikation ist der eigentliche Warping-Pfad jedoch nicht von Bedeutung, sondern nur die Kosten dieses Pfades, die gleichzeitig auch als warping-Distanz dienen. Diese können direkt aus der Abstandsmatrix DTW abgelesen werden.

$$warpDist(Q, R) := min\{DTW[n, j] \mid j \in \{1, 2, ..., m\}\} \tag{2.2}$$

Wird die Zeitreihe Q mit mehreren anderen Zeitreihen aus einer Referenzmenge S_R mittels DTW verglichen, gibt jeder Vergleich einen günstigsten Pfad zurück. Am Ende wird die Zeitreihe, die von allen Vergleichen die insgesamt minimalen Pfadkosten zurückgibt, die zur abgefragten Reihe ähnlichste Sequenz und liefert damit nach dem nearest-neighbour-Prinzip den Klassifikator.

$$bestFit(Q) := min\{warpDist(Q, R) \mid \forall R \in S_R\} \tag{2.3}$$

2.2.3 Problem mit der Zeit

Bisher wurde dargestellt, wie DTW Zeitreihen vergleicht und zur Klassifikation genutzt werden kann. Eine der Fragen, die in Abschnitt 1.3 gestellt wurden, war ob mit DTW auch zeitliche Dynamiken unterschieden werden können. Hier kann es aufgrund der Arbeitsweise von DTW zu Problemen kommen.

Da beim DTW die zu vergleichende Zeitreihe entlang der Zeitachse gestreckt bzw. gestaucht wird, kann sie auf eine ganze Vielzahl von anderen Sequenzen unterschiedlicher Länge abgebildet werden [BC94], vgl. Abbildung 2.3.

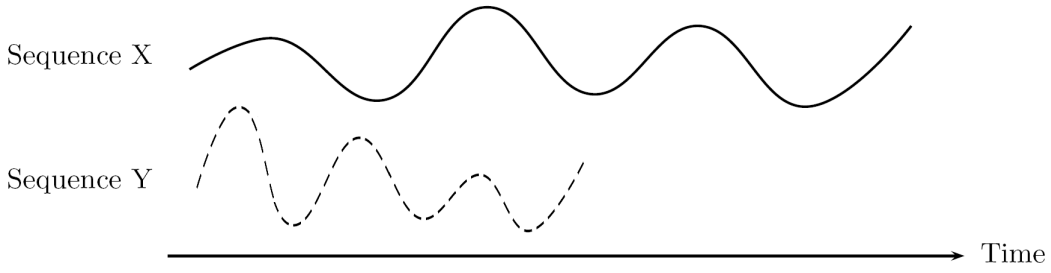

Abbildung 2.3: Sequenz Y ist zwar deutlich kürzer, könnte jedoch trotzdem erfolgreich auf Sequenz X abgebildet werden.

Aus diesem Grund könnte eine schnell ausgeführte Geste als eine langsam ausgeführte Geste klassifiziert werden, wenn sie durch DTW zu sehr entlang der Zeitachse gestreckt wird.

2.3 Verwandte Arbeiten

Nachdem die Funktionsweise von DTW erklärt wurde, wird nun im Folgenden dargestellt, wie DTW in der Vergangenheit zur Gestenerkennung genutzt wurde. Es wird auch ein Überblick über verschiedene andere Methoden gegeben, die ebenfalls zur Erkennung von Gesten eingesetzt wurden.

2.3.1 Gestenerkennung mittels DTW

DTW ist inzwischen eine verbreitete Methode zu Gestenerkennung, obwohl es ursprünglich für eine gänzlich andere Anwendung entwickelt wurde. Gemeint ist das Vergleichen von eindimensionalen Signalfolgen. Wie jedoch bereits in Abschnitt 2.1 erwähnt wurde, haben aufgezeichnete Gesten, je nach Anwendung, mehrere Dimensionen. Aus diesen Grund war ein Ausweiten von DTW auf den mehrdimensionalen Fall notwendig.

Mehrdimensionales DTW verläuft nahezu analog zum eindimensionalen Fall. Statt jedoch nur zwei eindimensionale Sequenzen zu vergleichen, werden die einzelnen Dimension paarweise betrachtet. Der wichtige Unterschied in den verschiedenen Verfahren des mehrdimensionalen DTW liegt darin, wie der Vergleich von mehr als einer Dimension verarbeitet werden soll. Prinzipiell gibt es zwei Möglichkeiten wie die Vergleiche von mehreren Dimensionen zusammengeführt werden können: das sogenannte abhängige bzw. unabhängige mehrdimensionale Dynamic Time Warping (MD-DTW) [SYHJ+16].

$$DTW_D(Q,R) = DTW(\{Q_x, Q_y, Q_z\}, \{R_x, R_y, R_z\}) \tag{2.4}$$

$$DTW_I(Q,R) = DTW(Q_x, R_x) + DTW(Q_y, R_y) + DTW(Q_z, R_z) \tag{2.5}$$

Beim abhängigen MD-DTW (DTW_D) wird der gleiche warp auf alle Dimensionen angewandt, während beim unabhängigen MD-DTW (DTW_I) jede Dimension individuell, ohne Einfluss der anderen Dimensionen, gewarpt wird. Beide Methoden sind für sich genommen sinnvoll und keine ist generell besser als die andere.

In der Vergangenheit wurde bereits Kinect eingesetzt um Gesten aufzuzeichnen und mit DTW zu verarbeiten. Dies wurde u.A. genutzt um mit Hilfe von Gesten mit Robotern zu interagieren [BDH13]. Eine Erweiterung des DTW-Ansatzes stellt das sogenannte probabilistische DTW dar. Dabei wird eine Sequenzen nicht direkt mit anderen Referenzsequenzen verglichen, sondern wie wahrscheinlich die Sequenz zu einer Gruppe von ähnlichen Referenzsequenzen gehört [BHVP+13].

DTW konnte ebenfalls auch schon erfolgreich zum Vergleich unvollständiger Zeitreihen eingesetzte werden [TGQS09]. Dabei handelte es sich jedoch um eine medizinische Anwendung, basierend auf EEG-Signalen, nicht jedoch um die Erkennung mehrdimensionaler Gesten.

Für die in dieser Arbeit untersuchte Frage, ob DTW geeignet ist, Gesten anhand ihrer Ausführungsgeschwindigkeit zu unterscheiden, gab es bisher keine Untersuchungen. Ein auf DTW basierendes System zur Erkennung von verschiedenen Geschwindigkeiten, speziell im Anwendungsfeld der Gestenerkennung, wurde unter anderem bereits in [FOF12] vorgeschlagen und dessen theoretische Möglichkeiten diskutiert.

Es wurde vermutet, dass DTW in der Lage wäre, Gesten mit unterschiedlichen Geschwindigkeiten zu erkennen, indem z.B. schneller ausgeführte Gesten natürlicherweise auf andere schnell ausgeführte Gesten mit einer geringeren DTW-Distanz abgebildet werden könnten. Da idealerweise jeder Geste sowohl mit einem Formals auch einem Geschwindigkeitslabel versehen wären, könnte so auch die Geschwindigkeit der abgefragten Geste ermitteltet werden. Jedoch wurde dies nicht weiter untersucht.

2.3.2 Weitere Methoden zur Gestenerkennung

Auf dem Feld der Gestenerkennung werden bereits eine Vielzahl an Methoden aus den Gebieten der Signalverarbeitung und des Machine Learning angewandt. Zu nennen wären da HMMs, Finite State Machines (FSM) oder Computational-Intelligence- bzw. Soft-Computing-Methoden [MA07].

HMMs werden nun schon seit über 20 Jahren erfolgreich zur Erkennung von Gesten genutzt. Sie bestehen aus einem Netz von nicht-beobachtbaren Zuständen, die zu bestimmten Zeitpunkten beobachtbare Symbole erzeugen, wie in Abbildung 2.4 dargestellt. Eine Klasse von Gesten wird durch ein Teil-HMM repräsentiert. Es wird dann bestimmt welches dieser Teil-HMMs am wahrscheinlichsten eine zu erkennende Geste erzeugt hat [NKW96].
Nach aktuellem Stand der Forschung sind HMMs in der Lage, Gesten anhand ihrer Ausführungsgeschwindigkeit zu unterscheiden [Mar17].

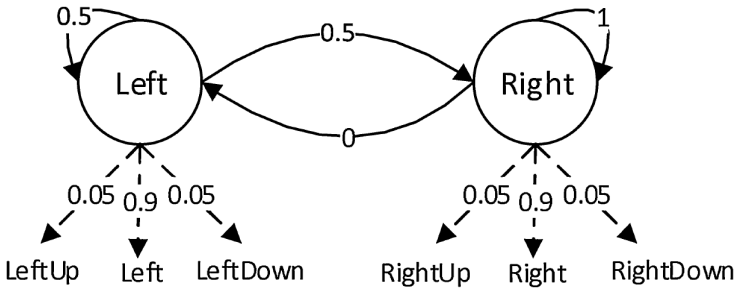

Abbildung 2.4: Ein HMM mit zwei versteckten Zuständen, das die Bewegung einer Hand von links nach rechts modelliert [DKH15].

FSMs sind relativ ähnlich zu HMMs, in der Hinsicht, dass auch hier Gesten als eine Folge von Zuständen modelliert werden, s. Abbildung 2.5. Im Gegensatz zu HMMs kommen FSMs jedoch auch mit weniger Trainingsdaten aus [HTH00a]. Die Erkennung von Gesten verläuft nach dem nahezu gleichen Prinzip wie bei HMMS. Jede Gestenart wird als eine Folge von Zuständen modelliert. Abhängig vom den Werten einer zu erkennenenden Gestensequenz, wird zu jedem Zeitpunkt entschieden, ob der Zustand gewechselt werden soll. Eine Geste gilt als erkannt, wenn einer der Endzustände der FSM erreicht wird [HTH00b].

Die Methoden aus dem Bereich der Computational Intelligence sind besonders interessant, aufgrund ihrer Fähigkeit mit Ungenauigkeit und Ungewissheit umgehen zu können [MA07].
Fuzzy Mengen können mit Ungenauigkeit umgehen und sind damit in der Lage,

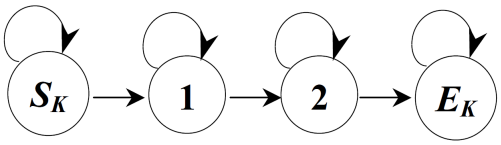

Abbildung 2.5: Eine FSM zur Erkennung einer Geste mit vier Zuständen [HTH00a].

Gesten nicht nur einer Klasse strikt zuzuordnenden, sondern auch Mehrfachzugehörigkeiten zu handhaben. Unter Verwendung von Fuzzy Regel konnten z.B. bereits Zeichensprachen verarbeitet werden [BCD06]. Dabei wurden mit Hilfe eines Data Gloves zuerst die Winkel zwischen einzelnen Fingergliedern genutzt, um die Konfiguration der Hand zu erkennen, um dann aus einer Folge von Konfigurationen die ausgeführte Geste zu bestimmen.
Neuronale Netze konnten ebenfalls genutzt werden, um Zeichensprachen zu verarbeiten [MT91]. Häufig wird dort ein Kamera-basierter Ansatz verfolgt, wobei mehrere Bilder von der Hand aufgenommen und verarbeitet werden. Davon kann dann die Form der Hand extrahiert und zur mittels eines Neuronalen Netzes zur Gestenerkennung genutzt werden [SP09]. Der große Vorteil dieses Ansatzes ist neben der hohen Erkennungsrate und Echtzeitfähigkeit, die Tatsache dass die Erkennung einer Geste unabhängig von ihrer Position, Rotation und Skalierung erfolgen kann [Ahm12].

3. Implementierung

Nachdem im letzten Kapitel die notwendigen Grundlagen erklärt wurde, werden in diesem Kapitel die genaueren Details der Implementierung erläutert. Zuerst wird erklärt welcher DTW-Algorithmus gewählt wurde und wie er funktioniert. Außerdem wird der genutzte Datensatz an Gesten beschrieben sowie verschiedene Ansätze zur Nachbildung unvollständiger Daten und zur Modellierung zeitlicher Dynamiken erklärt.

3.1 DTW-Algorithmus

Im Rahmen dieser Arbeit wurde die DTW-Erkennungsmethode von [tHRH07] als Grundlage genutzt. Die dort verwendete DTW-Methode ist abhängig, d.h. für alle Dimensionen wird der gleiche Warp verwendet. Damit wird erzwungen dass die gesamte Geste mit jeder Dimension für die Bestimmung der Ähnlichkeit betrachtet wird. Wird eine Testgeste abgefragt, wird sie mit allen abgespeicherten Referenzgesten verglichen, wobei jeder Vergleich einen DTW-Abstand zurück gibt. Die anschließende Klassifikation der abgefragten Geste verläuft nach dem nearest-neighbour-Prinzip, d.h. die Referenzsequenz mit dem laut Algorithmus geringstem DTW-Abstand bestimmt, als welche Klasse die abgefragte Testgeste klassifiziert wird.

Der Algorithmus kann mit verschiedenen Parametern konfiguriert werden. In den Experimenten werden später alle möglichen Konfigurationen getestet. Bei den genutzten Parametern handelt es sich um Normalisierung, Ableitungen und verschiedene Abstandsmaße.

Mit Normalisierung ist hier die z-Transformation der einzelnen Dimensionen gemeint. Für alle Werte X_i einer Dimension X wird der Mittelwert μ und die Standardabweichung σ berechnet.

$$\mu = \frac{1}{n} \sum_{i=1}^{n} X_i \tag{3.1}$$

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_i - \mu)^2} \tag{3.2}$$

Diese werden anschließend genutzt um X auf den Mittelwert 0 und die Standardabweichung 1 zu transformieren.

$$Z_i = \frac{X_i - \mu}{\sigma} \tag{3.3}$$

Zweck der Normalisierung ist es, eine Unabhängigkeit von Position und Skalierung der Geste zu erreichen. Der Mittelpunkt der Geste (μ) wird von einer beliebigen Position im Raum auf einen festen Nullpunkt gelegt. Die Ausmaße der Geste (σ) werden so ebenfalls standardisiert, sodass es keinen Unterschied mehr macht wie groß oder klein ein Proband eine Geste zeichnet. Zeitliche Dynamiken der Geste, wie die Ausführungsgeschwindigkeit, bleiben dagegen unverändert, weil z.B. zeitliche Abstände zwischen den Koordinaten oder die Länge der Gestensequenz nicht beeinflusst wird.

Der zweite Parameter, Ableitungen, bezeichnet die erste Ableitung zwischen zwei Koordinatenpunkten. Ein Punkt $\vec{p}_t$ ist dabei die aufgezeichnete Position der rechten Hand einer Person im Raum zum Zeitpunkt t. Die Ableitung des Punktes $\vec{p}_t$ ist dann die Differenz zwischen diesem Punkt und seinem nachfolgenden Punkt.

$$der(\vec{p}_t) = \vec{p}_{t+1} - \vec{p}_t \tag{3.4}$$

Die Ableitungen erfüllen einen ähnlichen Zweck wie die Normalisierung: Sie sollen eine Unabhängigkeit von der Position ermöglichen [KP01]. Die erste Ableitung einer Größe beschreibt die Veränderung dieser Größe in Abhängigkeit von ihren Parametern. In diesem Fall bedeutet dies die Veränderung der Position im Laufe der Zeit, während die tatsächliche Position ignoriert wird. Anders als bei der Normalisierung können Ableitungen jedoch keine Unabhängigkeit von der Skalierung der Geste ermöglichen.

Der dritte Parameter ist die Wahl eines Abstandsmaßes. Wie bereits in Abschnitt 2.2 dargestellt, benötigt DTW ein Abstandsmaß, um die Entfernung zwischen zwei Punkten beschreiben und damit die Ähnlichkeit von zwei Sequenzen berechnen zu können. Für die Experimente wurden der Manhattan-Abstand d_m, der euklidische Abstand d_e und der quadratische euklidische Abstand d_s gewählt.

$$d_m(\vec{q}_i, \vec{r}_j) = \sum_{l=1}^{D} |\vec{q}_{il} - \vec{r}_{jl}| \tag{3.5}$$

$$d_e(\vec{q}_i, \vec{r}_j) = \sqrt{\sum_{l=1}^{D}(\vec{q}_{il} - \vec{r}_{jl})^2} \tag{3.6}$$

$$d_s(\vec{q}_i, \vec{r}_j) = \sum_{l=1}^{D}(\vec{q}_{il} - \vec{r}_{jl})^2 \tag{3.7}$$

3.2 Genutzte Daten

Für die Experimente wurde ein Set von zehn Beispiel-Gesten erstellt, die in Abbildung 3.1 dargestellt sind. Insgesamt gibt es sechs verschiedene Gestenformen. Für zwei Gestenformen wurde jeweils zwei bzw. vier verschiedene zeitliche Dynamiken definiert. Jede der zehn Gesten wurde jeweils 20 mal von fünf verschiedenen Testpersonen durchgeführt und mittels Kinect aufgezeichnet, was in einen Satz von insgesamt 1000 aufgezeichneten Gesten resultiert.

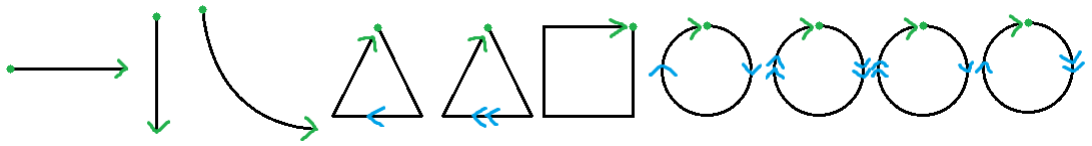

Abbildung 3.1: Übersicht des Gestensets. Grüne Markierungen zeigen Start- und Endpunkte der Geste, während blaue Pfeile Unterschiede in der Ausführungsgeschwindigkeit markieren.

Bei den ersten zwei Gesten handelt es sich um eine einfache horizontale und eine vertikale Linie. Danach folgt eine diagonal geschwungene Kurve. Sie dienen primär dazu, die Unterscheidung von Gestenformen zu untersuchen, da ihre Formen sich grundsätzlich von den anderen Gesten unterscheiden
Die nächsten zwei Gesten sind formgleiche Dreiecke, von denen das erste langsamer und das zweite schneller ausgeführt wird. Dann folgt ein Quadrat, das gewählt wurde, da seine Form, als eine in sich geschlossene Kurve, relativ ähnlich zu Kreisen und Dreiecken ist.
Bei den letzten vier handelt es sich um formgleiche Kreise mit verschieden Mustern in der Ausführungsgeschwindigkeit: ein langsamer und ein schneller Kreis, ein Kreis, der in der ersten Hälfte langsam und in der zweiten schnell gezeichnet wird, sowie ein Kreis, der in der ersten Hälfte schnell und der zweiten langsam gezeichnet wird.

3.3 Simulation unvollständiger Daten

Eine der Fragen in Abschnitt 1.3 war, ob DTW auch in der Lage ist, Gesten basierend auf einer unvollständigen Sequenz korrekt zu erkennen. Um dies experimentell zu überprüfen, wurden zwei Methoden, s. Abbildung 3.2, festgelegt, um unvollständige Sequenzen zu simulieren.

Abbildung 3.2: Zwei Methoden zur Simulation unvollständiger Daten. Nur rot markierte Frames werden für die Erkennung der Geste genutzt. Schwarz markierte Frames werden ignoriert.

Bei der ersten Variante, der reduzierten Bildrate, wurde nur jedes dritte aufgezeichnete Frame einer Geste verwendet. Diese reduzierte Sequenz entspreche dann einer Geste, die von einer Kamera mit nur einem Drittel der eigentlichen Bildrate aufgenommenen wurde.
Die zweite Variante, teilt die gesamte Sequenz in Gruppen von je fünf aufeinanderfolgenden Frames ein. Von diesen Gruppen wird dann nur jede zweite verwendet. Durch das Auslassen von mehren aufeinanderfolgenden Frames, sollen Unterbrechungen bzw. Aussetzer während der Aufzeichnung nachgebildet werden. In der Realität mögen solche Aussetzer zwar nicht mit einer derartigen Regelmäßigkeit auftreten, aber zum Zwecke der Vergleichbarkeit wurde hier auf eine zufällig gesteuerte Auswahl der Gruppen verzichtet.

3.4 Ansätze für zeitliche Dynamiken

Bisher wurde nur diskutiert, wie die aufgezeichneten Raumkoordinaten einer Geste zu verarbeiten sind. Es könnte jedoch sein, dass reine Raumkoordinaten zur Unterscheidung zeitlicher Dynamiken nicht ausreichen. Aus diesen Grund wird noch als zusätzliche vierte Dimension ein Zeitstempel eingeführt.

Neben den von Kinect erfassten Raumkoordinaten, wurde ebenfalls der Zeitpunkt, zu dem die Koordinaten aufgezeichnet wurden, gespeichert. Um besser mit den Zeiten arbeiten zu können wurden sie auf ein geeignetes Format umgeformt. Das erste aufgezeichnete Frame einer Geste hat den Zeitstempel 0. Alle darauffolgende Frames speichern den zeitlichen Abstand zum ersten Frame in Millisekunden.

Kinects Kamera zeichnet standardmäßig mit 30 Bildern pro Sekunde auf. Aus diesem Grund sollten im Idealfall alle aufgezeichneten Bilder einen zeitlichen Abstand von 33 Millisekunden haben. Aufgrund technischer Limitierungen sind diese zeitlichen Abstände jedoch nicht immer gleich groß, sondern streuen teilweise sehr stark, z.B. zwischen zehn und 50 Millisekunden. Um diese Streuung auszugleichen werden in einen zweiten Ansatz die Zeitstempel künstlich aus den festen zeitlichen Idealabstand von 33 Millisekunden festgelegt.

4. Experimentelle Verifikation

Nun, da im letzten Kapitel die Implementierung erklärt wurde, kann mit den Experimenten begonnen werden. Zunächst wird kurz der Ablauf der Experimente dargestellt und eine Übersicht über im Vorfeld aufgestellte Erwartungen und Hypothesen gegeben. Zu Beginn wird ein Grundtest durchgeführt, der zeigen soll, ob der implementierte Algorithmus Gesten erkennen kann und als Vergleich für weitere Experimente dient. Danach wird untersucht, ob die Erkennungsrate von Gesten durch fehlerhafte Daten beeinträchtigt werden kann. Zuletzt folgt dann die Untersuchung, in wie fern DTW in der Lage ist zeitliche Dynamiken zu unterscheiden.

4.1 Experimentablauf

Der Datensatz wird in einen Referenz- und einen Testanteil aufgeteilt. Jede Testgeste wird mittels DTW mit jeder Geste aus dem Referenzsatz verglichen. Jede aufgezeichnete Gestensequenz ist mit mehreren Labeln versehen, die u.a. die Form der Geste, die ausführende Person und, falls notwendig, das verwendete Geschwindigkeitsmuster festhalten. Die Label der laut DTW-Algorithmus ähnlichsten Sequenz werden dann mit tatsächlichen Labeln der Testsequenz verglichen. Am Ende wird der Anteil korrekt erkannten Label ermittelt.

Um statistisch sinnvolle Aussagen ableiten zu können, wurden die Experimente mit Kreuzvalidierung durchgeführt. Dabei wurde eine fünffache Kreuzvalidierung genutzt, d.h. jedes Experiment wurde fünf mal durchgeführt, wie in Abbildung 4.1 dargestellt ist. Dabei wurden die Daten in 80% Referenz- und 20% Testdaten eingeteilt und die Unterteilung wurde für jedes Experiment, nach einem festgelegten Muster, neu vorgenommen wurde. Bei den, in diesem Kapitel aufgelisteten, Ergebnissen handelt es sich daher um die Durchschnittswerte von jeweils fünf Experimenten.

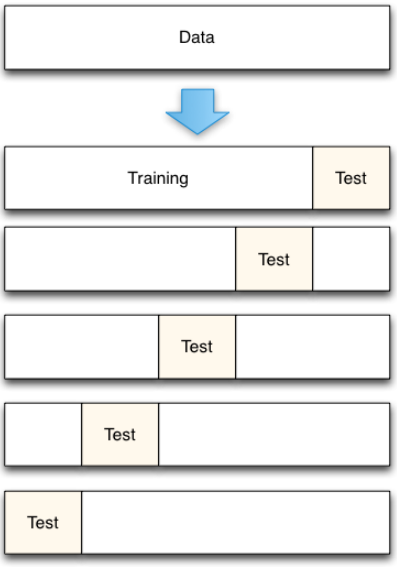

Abbildung 4.1: Schematische Darstellung des Testmengenwechsels bei fünffacher Kreuzvalidierung [cro]

Um genauere Aussagen treffen zu können wurden zusätzlich noch Konfusionsmatrizen erstellt, in denen festgehalten wurde, welche ausgeführte Geste wie klassifiziert wurde. Eine Auswahl dieser Matrizen ist im Anhang hinterlegt.

Jedes Experiment wurde außerdem mit den folgenden Parametern durchgeführt: Dem Nutzen von Ableitung, eine dimensionsspezifische Normalisierung der Daten sowie sie Wahl verschiedene Abstandsmaße. Im Laufe jeder Experimentreihe werden alle Kombinationen der Parameter nacheinander getestet werden.

4.2 Erwartungen

Im Vorfeld der Experimente wurden folgende Hypothesen aufgestellt:

1. Fehlerhafte bzw. unvollständige Daten verschlechtern die Erkennungsraten.
Durch Lücken in den Gestensequenzen würde die DTW-Distanz steigen, wodurch die Wahrscheinlichkeit einer Fehlklassifikation ebenfalls erhöht werden könnte.

2. Formgleiche Gesten mit unterschiedlichen zeitliche Dynamiken können nicht unterschieden werden.
DTW verzerrt Zeitreihen nichtlinear entlang der Zeitachse. Damit ist es möglich, dass schnell ausgeführte Gesten als langsam aufgeführte Gesten klassifiziert werden können.

3. Eine Identifikation von Personen aufgrund ausgeführter Gesten ist nicht möglich.
Auch dies kann mit der nichtlinearen Verzerrung durch DTW erklärt werden. Dadurch kann es schwerer sein individuelle Ausführungsmuster in Gesten wiederzuerkennen.

4.3 Untersuchung allgemeiner Gestenerkennung

Zunächst wurde ein grundlegender Test durchgeführt, der als Ausgangspunkt für die weiteren Experimente dienen soll. Zweck dieses ersten Tests war es, zu überprüfen, ob der implementierte Algorithmus überhaupt in der Lage ist, Gesten zu erkennen.

Die Ergebnisse sind in Tabelle 4.1 festgehalten. In der Spalte „Form“ stehen die Anteile der korrekt erkannten Gestenformen, in der Spalte „User“ die Anteile der korrekt wiedererkannten Probanden, wenn zugleich die Gestenform korrekt erkannt wurde. Zur Klassifizierung wurden sechs Klassen genutzt. In Abschnitt A.1 ist eine Auswahl der Konfusionsmatrizen für die Formerkennung hinterlegt.

Ableitungen	Normalisierung	Abstandsmaß	Gestenform	Nutzer
false	false	Manhattan	80,2584	99,0252
		Euklidisch	80,4574	98,8828
		Quadratisch	80,3788	98,8828
	true	Manhattan	79,1574	50,9586
		Euklidisch	79,1574	51,9602
		Quadratisch	79,4798	51,5554
true	false	Manhattan	33,0564	38,7796
		Euklidisch	31,9564	38,4774
		Quadratisch	33,6346	39,381
	true	Manhattan	66,2252	49,1678
		Euklidisch	60,6232	46,5652
		Quadratisch	59,3576	46,6348

Tabelle 4.1: Ergebnisse des Grundtests. Festgehalten wurden die Anteile der korrekt erkannten Gestenformen und der ausführenden Personen.

Die beste Erkennungsrate von 80% liegt vor, wenn die Gestensequenzen unverarbeitet verwendet werden. Wurde die Form der Geste korrekt erkannt, kann die Geste dann auch mit 99 prozentiger Sicherheit der richtigen Person zugeordnet werden. Wurden die Daten normalisiert, ist die Erkennungsrate nur minimal schlechter mit 79%. Dafür sinkt aber die Personenzuordnung auf 51%. Bei abgeleiteten Sequenzen sinkt die Formerkennung weiter runter auf 33% und die Personenzuordnung auf 39%. Wurden von den Gestensequenzen sowohl Ableitungen gebildet, als auch eine Normalisierung vorgenommen, schwankt die Formerkennungsrate zwischen 59 und 66%, während die ausführende Person in 46 bis 49% der Fälle erkannt werden konnte.
Die Wahl des Abstandsmaßes hat in fast allen Tests nur einen minimalen Einfluss auf beide Erkennungsraten.

Aus den Ergebnissen lässt sich schließen, dass der Algorithmus in der Lage ist Gesten, zumindest anhand ihrer Form, zu erkennen. Auffällig ist, dass die Formerkennung am Besten auf unverarbeiteten Daten abschneidet. Gleichzeitig konnte auch in fast allen Fällen die dazugehörige Person korrekt festgestellt werden. Dieses nahezu perfekte Ergebnis mag auf den ersten Blick positiv erscheinen, könnte jedoch eher das Resultat eines unerwünschten Effekts sein. Womöglich wurde nicht die allgemeine Form der Geste, sondern eher das individuelle Ausführungsmuster einer Person erkannt. Ein möglicher Grund dafür kann in der Art der Datengewinnung liegen. Dadurch,

dass die Probanden die gleiche Geste mehrmals hintereinander ausführen mussten, sind sie wahrscheinlich mit der Zeit in ein bestimmtes Muster verfallen, sodass einige Gesten nahezu vollständig identisch sein könnten. Dies sollte nachträglich genauer untersucht werden.
Ähnlich gut sind die Ergebnisse der zweiten Testreihe, in der die Sequenzen normalisiert wurden. Die Formerkennung ist nur minimal schlechter als bei den unverarbeiteten Sequenzen, gleichzeitig wurde aber die Personenerkennungsrate halbiert. Der Grund dafür liegt darin wie die Normalisierung funktioniert. Wie bereits in Abschnitt 3.1 erklärt, wird der Einfluss von Position und Skalierung aufgehoben. Während also die Form insgesamt gleich bleibt, sind hier individuelle Muster schwerer zu identifizieren.
Deutlich schlechter sind die Ergebnisse mit Ableitungen. Nur noch ein Drittel der Gestenformen konnte korrekt klassifiziert werden. Vermutlich ist durch die Ableitung die Kontinuität der Daten verringert wurden, z.B. durch Sprünge in den Sequenzen. Diese könnten die Erkennung der Gestenformen erschwert haben.

4.4 Untersuchung mit fehlerhaften Daten

Eine der im Vorfeld gestellten Fragen war: Kann der Algorithmus Gesten auch erkennen, wenn die zu untersuchende Sequenz unvollständig ist? Um dies zu untersuchen, wurden die Testdaten auf zwei Arten modifiziert, die bereits in Abschnitt 3.3 erklärt wurden. Im Gegensatz dazu wurden die Referenzdaten unverändert gelernt.

Die Ergebnisse sind in Tabelle 4.2 festgehalten. Anders als im Grundtest ist die Nutzererkennung hier nicht von Bedeutung. Bei den Werten handelt es sich jeweils um den Anteil der korrekt erkannten Gestenformen, zeitliche Dynamiken wurden hier noch nicht berücksichtigt. Eine Auswahl von Konfusionsmatrizen für Formerkennung mit der reduzierten Bildrate ist in Abschnitt A.2 und für Aussetzer in Abschnitt A.3 zu finden.

Ableitungen	Normalisierung	Abstandsmaß	red. Bildrate	Aussetzer
false	false	Manhattan	80,5792	80,2792
		Euklidisch	80,6792	80,4788
		Quadratisch	81,1798	80,3788
	true	Manhattan	79,9788	77,0782
		Euklidisch	79,6782	78,2814
		Quadratisch	79,5788	77,9804
true	false	Manhattan	36,9346	33,1286
		Euklidisch	37,938	32,0286
		Quadratisch	37,136	33,6346
	true	Manhattan	62,3628	59,1586
		Euklidisch	62,8628	57,9552
		Quadratisch	63,3648	57,2522

Tabelle 4.2: Ergebnisse mit fehlerhaften Daten

Die beste Erkennungsrate von 80% liegt vor, wenn die Gestensequenzen unverarbeitet verwendet werden. Wurden die Daten normalisiert, ist die Erkennungsrate nur

minimal schlechter mit 79% bei der reduzierten Bildrate bzw. 78% bei Aussetzern. Bei abgeleiteten Sequenzen sinkt die Formerkennung auf 33% bei Aussetzern, aber nur auf 37% bei der reduzierten Bildrate. Wurden von den Gestensequenzen sowohl Ableitungen gebildet, als auch eine Normalisierung vorgenommen, pendelt sich die Formerkennungsrate auf 62% bzw. 58% ein.
Die Wahl des Abstandsmaßes hat in fast allen Tests nur einen minimalen Einfluss auf die Erkennungsrate.

Man kann feststellen, dass die Ergebnisse im Groben ähnlich ausfallen wie im Grundtest. Offenbar ist DTW in der Lage auch stark reduzierte Sequenzen korrekt zu erkennen.
Auffällig sind die leicht besseren Ergebnisse mit reduzierter Bildrate, wenn Ableitungen verwendet wurden. Vermutlich werden durch das systematische Auslassen von Frames, Sprünge in der abgeleiteten Sequenz reduziert, sodass die Geste leichter auf eine ähnliche Geste abgebildet werden kann.
Ein weiterer Vorteil ist, dass durch die kürzere Sequenzlänge jeder Geste auch die Rechenzeit stark reduziert ist.

4.5 Untersuchung mit unterschiedlichen zeitlichen Dynamiken

Die eigentliche Kernfrage, die am Anfang dieser Arbeit gestellt wurde, war, ob DTW in der Lage ist formgleiche Gesten mit unterschiedlichen zeitlichen Dynamiken zu erkennen. In den bisherigen Experimenten wurde nur die Fähigkeit zur Formerkennung untersucht. Mit den folgenden Experimenten werden auch zeitliche Dynamiken eingebunden. Um ein möglichst unbeeinflusstes Resultat zu erzielen, werden zunächst separat nur dreiecksförmige bzw. nur kreisförmige Gesten gegeneinander getestet.

Die Ergebnisse dieses Experiments sind in Tabelle 4.3 zu finden. Anders als in den vorherigen Experimenten wurde hier nicht der Anteil der korrekt erkannten Gestenformen, sondern der tatsächlichen Gesten festgehalten. Das bedeutet, ein schnell gezeichneter Kreis, der als langsam gezeichneter Kreis klassifiziert wurde, gilt als inkorrekt klassifiziert. Sowohl für die Erkennung von Dreiecken als auch Kreisen sind Konfusionsmatrizen in Abschnitt A.4 zu finden.

Bei unverarbeiteten Daten konnte die Ausführungsgeschwindigkeit in 67% der Fälle bei Kreisen und für 84% der Dreiecke korrekt erkannt werden. Wurden die Daten normalisiert, sank die Erkennungsrate bereits auf 54 bzw. 76%. Ableitungen sind, wie bei früheren Experimenten auch, noch schlechter für die Erkennungsraten, die auf 32% für Kreise und im Mittel auf 58% für Dreiecke sinkt. Eine Normalisierung der Daten, zusätzlich zur Ableitung, hat nur einen minimalen Effekt.
Wie bereits bei den anderen Experimenten hat die Wahl des Abstandsmaßes kaum einen Einfluss.

Auf den ersten Blick sehen diese Ergebnisse recht vielversprechend aus. Besonders bei den Dreiecksformen werden Erkennungsraten von 84% erreicht. Dieses scheinbar gute Ergebnis relativiert sich jedoch, wenn man bedenkt, dass dort lediglich zwischen zwei Alternativen gewählt werden muss.

Ableitungen	Normalisierung	Abstandsmaß	Kreise	Dreiecke
false	false	Manhattan	65,75	83
		Euklidisch	67,25	84
		Quadratisch	67,25	84
	true	Manhattan	54,25	76
		Euklidisch	53,25	77
		Quadratisch	54	75,5
true	false	Manhattan	33,5	62
		Euklidisch	32	58
		Quadratisch	31,25	53,5
	true	Manhattan	35,25	56,5
		Euklidisch	34,5	56,5
		Quadratisch	32,5	57

Tabelle 4.3: einfache Grundtest. als traindata wurden nur circle bzw triangle genutzt

Anders als bei der bisherigen Experimenten, bei denen die Formerkennung im Vordergrund stand, hat hier die Normalisierung der Daten einen deutlich negativeren Einfluss auf die Erkennungsrate. Dies ist deshalb unerwartet, da durch die Normalisierung die eigentliche Beschaffenheit einer Geste nicht verändert wird, sondern lediglich ihre Translation und Skalierung.
Besonders schlecht schneiden abgeleitete Daten ab. Unabhängig davon, ob sie auch normalisiert wurden oder nicht, sind die Erkennungsraten teilweise nur minimal besser als die Ergebnisse eines zufälligen Ratens.

4.6 Form und zeitliche Dynamiken

Im den nächsten Experimenten werden die kreis- und dreiecksförmigen Gesten nicht nur gegen sich selbst, sondern gegen den gesamten Gestensatz getestet. Folglich wird hier sowohl die Erkennung der korrekten Form als auch der zeitlichen Dynamik gemessen. Für die erste Reihe an Experimenten wurden nur die normalen Raumkoordinaten verwendet, genau wie bei den bisherigen Experimenten auch. Für zwei weitere Experiment-Reihen wurden die Daten zusätzlich um die zwei Möglichkeiten, die bereits in Abschnitt 3.4 dargestellt wurden, erweitert.

Die Ergebnisse aller drei Experiment-Reihen sind in Tabelle 4.4 festgehalten. Gemessen wurde wieder der Anteil der korrekt erkannten Gesten, was sowohl Form als auch zeitliche Dynamik einschließt. Für alle drei Testreihen wurden, wie zuvor auch, Konfusionsmatrizen erstellt. Eine Auswahl davon ist im Anhang in Abschnitt A.5 bis Abschnitt A.7 hinterlegt.

Bei unverarbeiteten Daten konnten in der ersten und dritten Experiment-Reihe ungefähr 55% der Gesten korrekt erkannt werden. Lediglich bei der Nutzung von Zeitstempeln treten starke Unterschiede zwischen verschiedenen Abstandsmaßen auf. Normalisierte Daten erzielen leicht schlechtere Ergebnisse. 46% bei der Nutzung der reinen Raumkoordinaten, 48 bis 50% wenn Zeitstempel hinzugezogen werden.
Abgeleitete Daten liefern auch hier wieder durchgehend schlechte Resultate. 19% korrekt erkannte Gesten nur unter Nutzung der Raumkoordinaten sind bereits inakzeptabel; zieht man Zeitstempel hinzu, sinkt es sogar noch weiter auf nur 14%.

Ableit.	Normal.	Abstandsmaß	Raumkoord.	Zeitstempel	künstl. Zeit
false	false	Manhattan	54,8332	53,1666	55,1662
		Euklidisch	55,4998	46,833	56,333
		Quadratisch	55,6664	29,4998	56,8328
	true	Manhattan	46,1664	50,3332	51,9996
		Euklidisch	46	48,8332	48,9996
		Quadratisch	46,3332	48,6664	48,4996
true	false	Manhattan	21,9994	14,4998	16,833
		Euklidisch	18,9996	14,4998	15,6662
		Quadratisch	18,6664	14,1664	15,9994
	true	Manhattan	24,1662	20,833	0
		Euklidisch	21,4998	18,833	0
		Quadratisch	21,4998	21,1664	0

Tabelle 4.4: Ergebnisse der zeitlichen Dynamiken

Wurden die Daten sowohl normalisiert also auch abgeleitet, verbessern sie sich wieder leicht auf ungefähr 20%. Eine offensichtliche Ausnahme bildet der letzte Teil der dritten Experiment-Reihe, mit den künstlichen Zeitstempeln. Dort konnte keine einzige Geste erkannt werden. Die Ursache dafür wird nachfolgend noch erklärt.
Die Wahl des Abstandsmaßes hat erneut nur einen minimalen Einfluss, wobei aber bei abgeleiteten Daten leicht bessere Ergebnisse mit dem Manhattan-Abstand erreicht wurden. Auffällig ist, dass lediglich bei den einfachen Zeitstempeln mit unverarbeiteten Daten, der quadratische Abstand deutlich schlechtere Ergebnisse zeigt.

Geschwindigkeitserkennung ist sehr schlecht, teilweise sogar deutlich schlechter als eine zufällig Suche.
Verschiedene Zeitimplementierungen haben auch nur eine geringe Auswirkung

Die Tatsache, dass bei den künstlichen Zeitstempeln durchgehend keine einzige Geste erkannt wurde, liegt an der speziellen Kombination von Zeitstempeln, Normalisierung und Ableitung. Da alle künstlichen Zeitstempel den gleichen Abstand haben, ist die Ableitung der Zeitkoordinate aufeinanderfolgender Frames immer gleich Null, vgl. Formel 3.4. Zusätzlich hat eine Menge von Nullen stets sowohl einen Mittelwert, als auch eine Standardabweichung von Null. Versucht man also eine Folge von Nullen nach Formel 3.3 zu normalisieren, tritt unweigerlich ein Fehler auf, da die Division durch Null nicht definiert ist.

4.7 Diskussion der Ergebnisse

Die Ergebnisse des ersten Grundtests zeigen, dass neben der Form einer Geste unter bestimmten Umständen auch die ausführende Person korrekt erkannt werden kann. Fehlerhafte bzw. unvollständige Daten haben keinen negativen Effekt auf die Erkennung von Gestenformen. Unterscheidung von zeitlichen Dynamiken ist prinzipiell möglich, jedoch nicht sehr zuverlässig.

Teilweise gab es bei den Experimenten auch deutliche Unterschiede in der Rechenzeit. Für normale Erkennung der Gestenform wurden im Schnitt 560 Millisekunden

benötigt um eine Geste zu klassifizieren. Mit unvollständigen Daten sank die Rechenzeit aufgrund der verkürzten Sequenzlänge deutlich ab; auf 190 Millisekunden bei der reduzierten Bildrate bzw. 300 Millisekunden bei den Aussetzern. Bei der Unterscheidung von zeitlichen Dynamiken bei Kreisen und Dreiecken betrug die Klassifikationszeit im Mittel nur 240 bzw. 130 Millisekunden, was aber eher an der stark reduzierten Menge an Referenzgesten liegt. Die Erkennung sowohl von Form als auch zeitlicher Dynamik betrug mit 580 Millisekunden nur minimal länger als im Grundtest. Wurde zusätzlich die Zeit als vierte Dimension genutzt, stieg die Klassifizierungszeit auf 630 Millisekunden an.

5. Zusammenfassung

In dieser Arbeit wurde die Erkennung von Gesten im 3D-Raum mittels Dynamic Time Warping untersucht. Besonderes Augenmerk lag, neben dem Erkennen von Gesten, auf den Fragen, ob mit einer Geste auch die ausführende Person identifiziert werden kann, ob fehlerhafte Daten die Gestenerkennung beeinträchtigen und natürlich ob Gesten mit der gleichen Form aber unterschiedlichen zeitlichen Dynamiken unterschieden werden können.
In diesem Kapitel werden die Ergebnisse der Experimente, die zur Beantwortung der in Abschnitt 1.3 gestellten Fragen durchgeführt wurden, noch einmal zusammengefasst. Außerdem wird auf einige Einschränkungen und Probleme der Arbeit eingegangen und ein Ausblick für die Zukunft gegeben.

5.1 Zusammenfassung der Ergebnisse

Zuerst wurde ein einfacher Grundtest durchgeführt. Dabei wurde untersucht, wie gut verschiedene Gesten nur anhand ihrer Form erkannt werden können, und falls die Gestenform korrekt klassifiziert werden konnte, auch die Person, die die Geste ausgeführt hat, identifiziert werden konnte. Insgesamt konnten 80% der Gestenformen korrekt klassifiziert und zusätzlich auch in 98% der Fälle der Nutzer identifiziert werden.

Im zweiten Experiment stand die Frage im Mittelpunkt, ob die Erkennungsrate durch fehlerhafte bzw. unvollständige Daten negativ beeinflusst wird. Dazu wurden die Testdaten so manipuliert, als ob sie entweder mit einer geringeren Bildrate aufgenommen wurden oder ob es bei der Aufzeichnung zu Aussetzern gekommen war. Auch hier konnten 80% der Gestenformen korrekt erkannt werden.

Für das nächste Experiment wurden Gesten, die zwar die gleiche Form aber unterschiedliche zeitliche Dynamiken haben, gegeneinander getestet. Dies diente dazu feststellen zu können, ob Gesten mit gleichen Form aber unterschiedlichen Ausführungsgeschwindigkeiten unterschieden werden können. In einem Experiment mit vier Klassen konnten 65% der Gesten korrekt klassifiziert werden bzw. 84% in einem Experiment mit zwei Klassen.

Im letzten Experiment wurden dann sowohl die Form als auch die zeitliche Dynamik der Geste für die Klassifikation berücksichtigt. Hier konnten nur noch 55% der Gesten korrekt klassifiziert werden.

5.2 Interpretation der Ergebnisse und Vergleich mit Erwartungen und Zielen

Die erste Erwartung war, dass eine ausgeführte Geste nicht korrekt einer Person zugeordnet werden kann. Die durchgeführten Untersuchungen deuten aber auf das Gegenteil hin. Teilweise konnten Erkennungsraten von 98% erreicht werden, wenn auch nur unter bestimmten Bedingungen.

Als nächstes wurde angenommen, dass fehlerhafte bzw. unvollständige Daten, die Erkennungsraten verschlechtern. Auch diese Annahme stellte sich als falsch heraus Sowohl eine geringere Bildrate als auch Aussetzer bei der Aufzeichnung hatten, entgegen der Erwartungen, keinen negativen Effekt auf die Erkennungsraten.

Die letzte Erwartung war, das Gesten nicht anhand ihrer zeitlichen Dynamiken unterschieden werden können. Dies wurde aufgrund der Arbeitsweise von DTW vermutet, bei der Sequenzen nicht-linear entlang der Zeitachse zu verzerrt werden. Diese Vermutung konnte mit den durchgeführten Experimenten bestätigt werden. Die Untersuchungen deuten darauf hin, dass die Unterscheidung von Gesten aufgrund ihrer Ausführungsgeschwindigkeit zwar ansatzweise möglich, aber für eine reale Anwendung nicht zuverlässig genug ist.

Damit konnten alle Ziele der Arbeit erreicht werden.

5.3 Einschränkungen

Ein Problem der Untersuchungen war die Nutzung eines recht eingeschränkten Testdatensatzes. Es wurden nur Daten von fünf Testpersonen verwendet, die hauptsächlich aus einfachen zweidimensionalen Gestenformen bestanden. Außerdem wurden die Daten für jede Person nur innerhalb einer Sitzung aufgezeichnet. Dies könnte dazu geführt haben, dass die Probanden in individuelle Ausführungsmuster beim Zeichnen der Gesten verfallen sind, was einen recht homogenen Datensatz zur Folge haben kann.

5.4 Erweiterungsmöglichkeiten

Die Ergebnisse der Nutzeridentifizierung sollten weiter untersucht werden. Wie bereits erklärt, kann der genutzte Gestensatz zu homogen ausgefallen sein. Daher wäre eine Folgeuntersuchungen angebracht, bei der die gleichen Personen die Gesten noch einmal ausführen. Sollten diese neuen Gesten ebenfalls richtig zugeordnet werden, kann eine Fehlerquelle ausgeschlossen werden und die Ergebnisse wären valider.
Spätere Nachforschungen haben ergeben, dass DTW bereits erfolgreich für eine gestenbasierte Nutzeridentifizierung genutzt werden konnte [GSL+14]. Im dort verwendeten Ansatz wurden alle Gesten eines Typs zur einer sogenannten „Super-Geste“

zusammengefasst. Dabei handelt es sich um einen allgemeinen Prototypen, der dann für die Gestenerkennung mit DTW genutzt wurde.

Eine weitere Möglichkeit wäre die Wahl einer anderen Klassifizierungsmethode. Die hier verwendete Nearest-Neighbor-Klassifikation ist ein einfaches und nur wenig robustes Verfahren. Wenn beim DTW-Vergleich eine nicht passende Sequenz zufälligerweise das beste Ergebnis erzielt, entsteht so eine Fehlklassifikation. Ein erster Schritt wäre z.B. das Nutzen von k-Nearest-Neighbor oder anderer, weiter entwickelter, Klassifizierungsverfahren.

5.5 Einsatzmöglichkeiten und Nutzen

Aufgrund der experimentellen Ergebnisse und nachträglichen Bestätigung durch [GSL+14] ist eine auf DTW basierte Gestenerkennung auch für eine Anwendung zur Nutzeridentifikation geeignet. Eine Gestenerkennung mittels DTW ist auch mit unvollständigen Daten möglich, was dieses System bedeutend robuster macht als bisher angenommen. Die Experimente haben außerdem gezeigt, dass DTW nicht geeignet ist, um Gesten anhand ihrer zeitlichen Dynamiken, z.B. in Form von verschiedenen Ausführungsgeschwindigkeiten, zu unterscheiden. Sollte dies für eine Anwendung notwendig sein, sollten daher andere Methoden in Betracht gezogen werden.

A. Anhang

Abkürzung	Bedeutung
L-v	vertikale Linie
L-h	horizontale Linie
L-d	diagonal geschwungene Kurve
T	Dreiecksform
T-s	langsam ausgeführtes Dreieck
T-f	schnell aufgeführtes Dreieck
S	Quadrat
C	Kreisform
C-s	langsam ausgeführter Kreis
C-f	schnell ausgeführter Kreis
C-sf	beschleunigter Kreis
C-fs	abgebremster Kreis

A.1 Grundtest

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	78	-	2	8	1	11
	L-h	1	90	4	-	-	5
	L-d	1	1	92	3	1	1
	T	8	1	1	137	7	46
	S	-	-	1	8	81	10
	C	10	2	1	51	12	324

Tabelle A.1: Ableitung = false, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	81	-	4	9	2	4
	L-h	1	97	-	-	-	2
	L-d	2	-	88	-	7	2
	T	7	-	1	135	-	57
	S	1	-	11	1	78	9
	C	6	2	8	58	14	312

Tabelle A.2: Ableitung = false, Normalisierung = true, Abstand = Manhattan

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	11	16	6	31	13	23
	L-h	10	26	7	18	12	27
	L-d	10	10	26	17	9	27
	T	9	23	-	60	19	63
	S	6	15	4	26	22	27
	C	42	50	7	96	45	160

Tabelle A.3: Ableitung = true, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	88	-	5	3	1	3
	L-h	2	91	2	1	1	3
	L-d	1	-	96	-	1	1
	T	-	-	-	89	13	98
	S	4	-	1	27	23	35
	C	-	1	6	130	37	226

Tabelle A.4: Ableitung = true, Normalisierung = true, Abstand = Manhattan

A.2 Fehlerhafte Daten - reduzierte Bildrate

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	76	-	2	9	1	12
	L-h	1	88	4	-	-	7
	L-d	1	1	93	2	1	1
	T	9	-	1	138	7	45
	S	-	-	2	8	80	10
	C	10	2	1	47	10	330

Tabelle A.5: Ableitung = false, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	80	-	3	10	5	2
	L-h	-	96	-	-	-	4
	L-d	1	-	89	-	7	2
	T	8	-	1	128	-	63
	S	-	-	10	-	82	8
	C	5	1	6	49	15	324

Tabelle A.6: Ableitung = false, Normalisierung = true, Abstand = Manhattan

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	17	7	10	18	15	33
	L-h	6	12	9	14	14	45
	L-d	14	6	43	11	5	20
	T	13	12	7	80	20	68
	S	7	8	10	13	21	41
	C	17	24	19	107	37	196

Tabelle A.7: Ableitung = true, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	81	-	16	-	-	3
	L-h	9	87	1	-	2	1
	L-d	-	2	94	-	-	3
	T	-	1	-	90	15	94
	S	3	1	1	29	29	37
	C	-	5	7	111	35	242

Tabelle A.8: Ableitung = true, Normalisierung = true, Abstand = Manhattan

A.3 Fehlerhafte Daten - Ausetzer

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	78	-	2	8	1	11
	L-h	1	90	4	-	-	5
	L-d	1	1	92	3	1	1
	T	8	1	1	137	7	46
	S	-	-	1	8	81	10
	C	10	2	1	51	12	324

Tabelle A.9: Ableitung = false, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	77	-	2	10	6	5
	L-h	2	96	-	-	-	2
	L-d	-	-	87	-	9	3
	T	12	-	-	121	-	67
	S	-	-	12	1	77	10
	C	5	2	9	63	9	312

Tabelle A.10: Ableitung = false, Normalisierung = true, Abstand = Manhattan

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	11	16	6	31	13	23
	L-h	10	26	7	18	12	27
	L-d	10	10	26	17	9	27
	T	9	23	-	86	19	63
	S	6	15	4	26	22	27
	C-s	42	50	7	96	45	160

Tabelle A.11: Ableitung = true, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste					
		L-v	L-h	L-d	T	S	C
Ausgeführte Geste	L-v	77	-	15	-	2	6
	L-h	3	95	1	-	-	1
	L-d	2	1	94	-	-	2
	T	-	1	-	77	20	102
	S	1	-	-	42	20	37
	C-s	-	-	3	132	37	228

Tabelle A.12: Ableitung = true, Normalisierung = true, Abstand = Manhattan

A.4 Zeitliche Dynamiken - nur Dreiecke und Kreise

		Klassifizierte Geste					
		T-s	T-f	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	81	19				
	T-f	15	85				
	C-s			73	11	5	11
	C-f			10	57	16	17
	C-sf			3	27	65	5
	C-fs			16	14	2	68

Tabelle A.13: Ableitung = false, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste					
		T-s	T-f	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	75	25				
	T-f	23	77				
	C-s			51	13	8	28
	C-f			17	48	24	11
	C-sf			13	21	54	12
	C-fs			21	7	8	64

Tabelle A.14: Ableitung = false, Normalisierung = true, Abstand = Manhattan

		Klassifizierte Geste					
		T-s	T-f	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	54	46				
	T-f	30	70				
	C-s			41	16	32	11
	C-f			29	31	27	13
	C-sf			27	15	40	18
	C-fs			31	20	27	22

Tabelle A.15: Ableitung = true, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste					
		T-s	T-f	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	33	67				
	T-f	20	80				
	C-s			25	23	35	17
	C-f			14	45	25	16
	C-sf			17	21	43	19
	C-fs			10	34	28	28

Tabelle A.16: Ableitung = true, Normalisierung = true, Abstand = Manhattan

A.5 Zeitliche Dynamiken

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	6	1	1	60	12	3	2	2	8	5
	T-f	2	-	-	10	55	4	10	5	4	10
	C-s	1	-	1	1	11	5	61	7	4	9
	C-f	1	-	-	7	6	3	7	49	14	13
	C-sf	8	1	-	3	6	-	2	23	55	2
	C-fs	-	1	-	6	11	4	15	13	1	49

Tabelle A.17: Ableitung = false, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	3	-	-	61	20	-	4	-	1	11
	T-f	4	-	1	15	39	-	14	8	13	6
	C-s	1	-	2	4	16	2	41	10	8	16
	C-f	-	1	3	-	9	5	14	38	21	9
	C-sf	1	1	1	2	15	3	7	14	46	10
	C-fs	4	-	2	5	7	4	15	6	5	52

Tabelle A.18: Ableitung = false, Normalisierung = truee, Abstand = Manhattan

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	3	19	-	26	12	10	9	6	7	8
	T-f	6	4	-	8	40	9	7	11	12	3
	C-s	16	8	2	15	8	17	10	10	13	1
	C-f	12	13	-	7	11	10	8	21	14	4
	C-sf	9	16	5	10	9	9	1	10	25	6
	C-fs	5	13	-	16	20	9	8	10	9	10

Tabelle A.19: Ableitung = true, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	-	-	-	17	25	7	10	18	15	8
	T-f	-	-	-	10	37	6	5	19	14	9
	C-s	-	1	5	11	19	11	13	10	20	10
	C-f	-	-	1	6	25	10	6	30	14	8
	C-sf	-	-	-	10	23	4	10	12	31	10
	C-fs	-	-	-	9	27	12	4	19	12	17

Tabelle A.20: Ableitung = true, Normalisierung = true, Abstand = Manhattan

A.6 Zeitliche Dynamiken - Zeitstempel

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	3	-	3	60	7	2	3	3	13	6
	T-f	5	-	-	10	58	5	3	4	3	12
	C-s	2	2	2	-	10	4	54	6	7	13
	C-f	2	-	2	3	5	5	7	44	17	15
	C-sf	13	1	-	5	3	-	1	18	56	3
	C-fs	9	1	-	4	16	5	5	13	-	47

Tabelle A.21: Ableitung = false, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	2	-	-	76	9	-	5	-	-	8
	T-f	4	-	1	14	33	-	16	9	13	10
	C-s	1	-	1	6	9	4	56	7	5	11
	C-f	-	1	2	1	8	5	21	35	19	8
	C-sf	-	-	-	3	11	4	13	9	53	7
	C-fs	2	-	2	6	4	6	22	3	6	49

Tabelle A.22: Ableitung = false, Normalisierung = true, Abstand = Manhattan

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	5	17	2	26	12	12	6	3	13	4
	T-f	6	19	1	4	14	18	9	13	9	7
	C-s	2	23	4	8	10	16	8	3	14	12
	C-f	7	16	2	5	6	13	15	13	15	8
	C-sf	4	18	3	11	11	12	8	7	17	9
	C-fs	8	16	4	6	16	14	5	9	13	9

Tabelle A.23: Ableitung = true, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	-	1	1	11	23	9	17	12	16	10
	T-f	-	-	-	6	28	16	19	9	15	7
	C-s	-	-	4	5	18	10	18	11	24	10
	C-f	-	-	3	9	13	11	23	16	15	10
	C-sf	-	-	1	9	7	10	13	8	37	15
	C-fs	-	1	1	6	13	18	18	15	13	15

Tabelle A.24: Ableitung = true, Normalisierung = true, Abstand = Manhattan

A.7 Zeitliche Dynamiken - künstliche Zeitstempel

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	6	1	1	63	11	2	2	1	8	5
	T-f	1	-	-	12	56	4	8	4	3	12
	C-s	1	-	1	1	9	4	59	9	6	10
	C-f	1	-	2	4	6	4	6	49	15	13
	C-sf	8	1	-	3	6	-	2	23	55	2
	C-fs	-	1	-	7	11	3	14	13	2	49

Tabelle A.25: Ableitung = false, Normalisierung = false, Abstand = Manhattan

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	2	-	-	76	9	-	5	-	-	8
	T-f	4	-	1	15	33	-	16	9	13	9
	C-s	1	-	1	7	8	4	63	4	4	8
	C-f	-	1	2	1	10	5	20	35	18	8
	C-sf	-	-	-	2	12	4	12	9	53	8
	C-fs	2	-	2	6	4	6	21	2	5	52

Tabelle A.26: Ableitung = false, Normalisierung = true, Abstand = Manhattan

		Klassifizierte Geste									
		L-v	L-h	L-d	T-s	T-f	S	C-s	C-f	C-sf	C-fs
Ausgeführte Geste	T-s	2	23	-	20	11	9	16	5	9	5
	T-f	9	14	1	10	24	7	12	6	12	5
	C-s	9	22	1	9	8	11	12	9	15	4
	C-f	2	15	-	11	15	12	16	16	9	4
	C-sf	6	16	5	11	8	12	5	11	21	5
	C-fs	2	22	-	8	16	10	16	9	9	8

Tabelle A.27: Ableitung = true, Normalisierung = false, Abstand = Manhattan

Literaturverzeichnis

[Ahm12] Tasnuva Ahmed. A neural network based real time hand gesture recognition system. *International Journal of Computer Applications*, 59(4):17–22, December 2012. (zitiert auf Seite 10)

[BC94] Donald J. Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, AAAIWS'94, pages 359–370. AAAI Press, 1994. (zitiert auf Seite 7)

[BCD06] Benjamín C. Bedregal, Antônio C. R. Costa, and Graçaliz P. Dimuro. *Fuzzy Rule-Based Hand Gesture Recognition*, pages 285–294. Springer US, Boston, MA, 2006. (zitiert auf Seite 1 und 10)

[BDH13] Saša Bodiroža, Guillaume Doisy, and Verena Vanessa Hafner. Position-invariant, real-time gesture recognition based on dynamic time warping. In *Proceedings of the 8th ACM/IEEE International Conference on Human-robot Interaction*, HRI '13, pages 87–88, Piscataway, NJ, USA, 2013. IEEE Press. (zitiert auf Seite 8)

[BHVP+13] Miguel Ángel Bautista, Antonio Hernández-Vela, Victor Ponce, Xavier Perez-Sala, Xavier Baró, Oriol Pujol, Cecilio Angulo, and Sergio Escalera. *Probability-Based Dynamic Time Warping for Gesture Recognition on RGB-D Data*, pages 126–135. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. (zitiert auf Seite 8)

[BKH11] Sascha Bosse, Claudia Krull, and Graham Horton. Modeling of gestures with differing execution speeds: Are hidden non-markovian models applicable for gesture recognition. In *Proceedings of the 10th International Conference on Modelling Applied Simulation (MAS)*, 12th-14th September 2011. (zitiert auf Seite 2)

[cro] Accurately measuring model prediction error. http://scott.fortmann-roe.com/docs/MeasuringError.html. Accessed on 20th February 2017. (zitiert auf Seite 16)

[DKH15] Tim Dittmar, Claudia Krull, and Graham Horton. A new approach for touch gesture recognition: Conversive hidden non-markovian models. *Journal of Computational Science*, 10:66 – 76, 2015. (zitiert auf Seite 2 und 9)

[FOF12] Tobias Franke, Manuel Olbrich, and Dieter W. Fellner. A flexible approach to gesture recognition and interaction in x3d. In *Proceedings of the 17th International Conference on 3D Web Technology*, Web3D '12, pages 171–174, New York, NY, USA, 2012. ACM. (zitiert auf Seite 8)

[GSL+14] Jože Guna, Emilija Stojmenova, Artur Lugmayr, Iztok Humar, and Matevž Pogačnik. User identification approach based on simple gestures. *Multimedia Tools and Applications*, 71(1):179–194, 2014. (zitiert auf Seite 24 und 25)

[HTH00a] Pengyu Hong, Matthew Turk, and Thomas S. Huang. Constructing finite state machines for fast gesture recognition. In *in Proc. 15th International Conference on Pattern Recognition (ICPR '00*, pages 691–694, 2000. (zitiert auf Seite 9 und 10)

[HTH00b] Pengyu Hong, Matthew Turk, and Thomas S. Huang. Gesture modeling and recognition using finite state machines. In *In Proceedings of the Fourth IEEE International Conference and Gesture Recognition*, 2000. (zitiert auf Seite 9)

[KP01] Eamonn J. Keogh and Michael J. Pazzani. Derivative dynamic time warping. In *In SIAM International Conference on Data Mining*, 2001. (zitiert auf Seite 12)

[KSa] Kinect sdk 1.0 - 3 - track bodies with the skeletonstream. http://archive.renauddumont.be/post/2012/04/19/Kinect-SDK-10-3-Track-bodies-with-the-SkeletonStream. Accessed on 9th January 2017. (zitiert auf Seite 5)

[LJ09] Zhi Li and Ray Jarvis. R.: Real time hand gesture recognition using a range camera. In *In: Australasian Conf. Robot. and Automat. (ACRA). Proc. on*, pages 529–534, 2009. (zitiert auf Seite 1)

[MÖ7] Meinard Müller. *Information Retrieval for Music and Motion*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007. (zitiert auf Seite 1 und 6)

[MA07] Sushmita Mitra and Tinku Acharya. Gesture recognition: A survey. *IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS - PART C*, 37(3):311–324, 2007. (zitiert auf Seite 1 und 9)

[Mar17] Suryakiran Maruvada. 3-d hand gesture recognition with different temporal behaviors using hmm and kinect. Master's thesis, Otto-von-Guericke-Universität Magdeburg, 2017. (zitiert auf Seite 2 und 9)

[MT91] Kouichi Murakami and Hitomi Taguchi. Gesture recognition using recurrent neural networks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '91, pages 237–242, New York, NY, USA, 1991. ACM. (zitiert auf Seite 10)

[NKW96] Yanghee Nam, Taejeon Korea, and KwangYun Wohn. Recognition of space-time hand-gestures using hidden markov model, 1996. (zitiert auf Seite 1 und 9)

[PMS+09] Farid Parvini, Dennis McLeod, Cyrus Shahabi, Bahareh Navai, Baharak Zali, and Shahram Ghandeharizadeh. *An Approach to Glove-Based Gesture Recognition*, pages 236–245. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. (zitiert auf Seite 1)

[SFC+11] Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *In In CVPR, 2011. 3*, 2011. (zitiert auf Seite 6)

[SP09] E. Stergiopoulou and N. Papamarkos. Hand gesture recognition using a neural network shape fitting technique. *Eng. Appl. Artif. Intell.*, 22(8):1141–1158, December 2009. (zitiert auf Seite 10)

[SYHJ+16] Mohammad Shokoohi-Yekta, Bing Hu, Hongxia Jin, Jun Wang, and Eamonn Keogh. Generalizing dtw to the multi-dimensional case requires an adaptive approach. *Data Mining and Knowledge Discovery*, pages 1–31, 2016. (zitiert auf Seite 8)

[TGQS09] Paolo Tormene, Toni Giorgino, Silvana Quaglini, and Mario Stefanelli. Matching incomplete time series with dynamic time warping: An algorithm and an application to post-stroke rehabilitation. *Artif. Intell. Med.*, 45(1):11–34, January 2009. (zitiert auf Seite 8)

[tHRH07] Gineke A. ten Holt, Marcel J.T. Reinders, and Emile A. Hendriks. Multi-dimensional dynamic time warping for gesture recognition. In *Thirteenth annual conference of the Advanced School for Computing and Imaging*, June 13-15 2007. (zitiert auf Seite 1 und 11)

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Magdeburg, den 19. März 2017