Adaptive Gestenerkennung mit Variationsabschätzung für interaktive Systeme

Maxim Boianetchii[1] und Marian Stein[2]

[1] Universität Rostock
maxim.boianetchii@uni-rostock.de
[2] Universität Rostock
marian.stein@uni-rostock.de

1 Einführung

Dieser Artikel stellt eine Gestenerkennung / ein Adaptionssystem für die Mensch-Computer-Interaktion vor. Es beschreibt ein vorlagenbasiertes Erkennungsverfahren, das durch einen Anpassungsprozess, die Gestenvariation in Echtzeit verfolgt. Die wichtigsten Vorteile sind geschätzten Parameter und Erkennungsergebnisse. Die Technik wurde auf verschiedene Weisen in einer Benutzerstudie mit 3D Gesten im freien Raum ausgewertet. Hierbei wurde festgestellt, dass das Verfahren robust gegen Rauschen ist und erfolgreich sich an Variationsparameter anpasst. Dieses Verfahren führt die Erkennung genauso gut oder sogar besser als andere Methoden durch.

Die Gesten werden für verschiedenen Formen der Aktivitätserkennung verwendet. Es besteht ein Bedarf für aufwändige Interaktionsparadigmen, die auf Körperbewegungen oder greifbare Geräte und Schnittstellen basieren. Es gibt bereits Verfahren zur diskreten und kontinuierliche Gestenerkennung, die erfolgreich umgesetzt wurden. Diese Verfahren wurden entwickelt um die Geste in den meisten Fällen zu markieren. In diesem Experiment wird eine ganz neue Technik vorgeschlagen, die über Klassifikation hinausgeht, eine Bewegung kennzeichnet und die Möglichkeit bietet, innovative Interaktionsszenarien zu nutzen. Die Benutzer müssen ständig ihre Bewegungen anzupassen und sich zum Beispiel auf optische oder akustische Rückmeldung verlassen. Die Gesten, die erkannt werden, können in Vergleich zu Referenzgesten „verzerrt“ angezeigt werden.Ein robustes Erkennnungssystem muss in der Lage sein, sich an solche Änderungen anzupassen. Außerdem wäre es nützlich die Parameter der Geste während der Ausführung schrittweise zu schätzen. Dies ermöglicht dem System die Berücksichtigung von Schwankungen, die während der Bewegung und Aktualisierung des Bewegungsmodells auftreten, wahrzunehmen. Derartige Parametrisierung könnte direkt in der Gestaltung der Interaktion verwendet werden. Caramiaux et al. [2] schlägt ein neues Gestenerkennungssystem vor, das in der Lage ist, in Echtzeit die berücksichtigte Bewegung und Variation inkrementell während der Durchführung wahrzunehmen und dem Nutzer Parameter zurückzuliefern. Es können verschiedene Gestenvariationen innerhalb der Geschwindigkeit, Amplitude oder Orientierung aufgenommen werden. Wichtig ist, dass diese Variationen kontinuierlich während der Ausführung der Gesten geschätzt werden können. Diese

Methode ist ein Zustandsraummodel, in dem Variationen online geschätzt werden. Dazu braucht man einen Partikelfilter um kontinuierlich die Variation der Geste anzupassen.

2 Ähnliche Arbeiten

In diesem Abschnitt untersuchen wir Methoden die am meisten benutzt werden um Gesten, die als mehrdimensionale Zeitreihen für die Mensch-Computer-Interaktion vorgestellt sind, zu erkennen. Die mehrdimensionalen Zeitreihen stellen die Bahn eines Punktes auf einer Oberfläche oder in dem 3D-Raum dar. Rubine[4] schlägt ein geometrisches Abstandmaß basierend auf Beispielen von Einzeltakt-Gesten vor. Wobbrock et al.[12] schlagen eine einfache Template-basierte Methode vor, die Nutzung des euklidische Abstands nach einer Vorverarbeitungsstufe zur Berücksichtigung geometrischer Variationen (wie Skalierung und Rotation) und Drehzahlschwankungen ermöglicht. [Gawrila und Davis[3] und Liu et al.[6] schlagen Dynamic Time Warping (DTW) vor, das erfordert die Speicherung des gesamten zeitlichen Struktur der Geste. Es gibt verschiedene Anwendungen, wie Gestensteuerung[8, Merrill et al.] kommunikative Gestenfolgen [1, Heloir et al. 2006], und Abfragen auf Basis menschlicher Bewegung [5, Forbes und Fiume 2005]. Statistische Methoden, wie das Hidden Markov Model (HMM) [10, Rabiner 1989], das auf einer Wahrscheinlichkeitsinterpretation der Beobachtungen (Probegesten) basiert und zeitliche Trajektorie der Geste durch eine kompakte Darstellung modellieren kann. Variationen in der Geste werden zum größten Teil durch Methoden wie HMM, die umfassende Datenbanken verwenden, unter Berücksichtigung aller möglichen Variationen. Wilson und Bobick[11] schlagen ein Modell vor, das die parametrische Änderungen in der Ausführung berücksichtigt. In diesem Artikel wird gezeigt, dass ein adaptiver Ansatz mit einem ausgefahrenen Zustand Modell und einem anderen Decodierschema möglich ist. Die von Caramiaux et al.[2] vorgeschlagene Methode, durch die Arbeit von Black und Jepson [7, 1998a] inspiriert, basiert auf dem Kondensationsalgorithmus [9, Isard und Blake 1998] für die Anerkennung der raumzeitlichen Vorlagegesten, der die Umsetzung für die Verfolgung von Geschwindigkeit und Skalierungsänderung erlaubt.Caramiaux et al.[2] verallgemeinern den Ansatz von Black und Jepson [1998a] durch Schätzen nicht nur der Skalierung, sondern auch anderer Parameter, wie Rotation.

3 Interaktionsprinzipien

Caramiaux et al.[2] präsentieren Interaktionsprinzipien, die wichtig für Zielanwendungen wie Ton Manipulation oder Visual Processing in Kontexten wie Spiele, interaktive Kunst oder Rehabilitation sind. Ihr Interaktionsmodel beinhaltet zwei Arten von Kontrolle: die Ausführung einer Geste und wie sie durchgeführt wird. Die Körperbewegungen werden von einem Bewegungserfassungsgerät erfasst. Der Algorithmus ist in der Lage die Geste zu erkennen, an ihre Variation

und Veränderungen anzupassen und die Variationsparameter mit Berücksichtigung der zuvor gelernten Vorlagen zurückliefern. Die Erkennung und Adaption werden in Echtzeit durchgeführt. Die Ausgabe des Algorithmus wird fortlaufend aktualisiert. Der Modelausgang hat zwei Komponenten: einen Index für die erkannten Gesten und einen Vektor für kontinuierliche Werte der Schätzung der Variation. Zur Erleichterung der schnellen Testsitzungen und um das Lernverfahren so leicht wie möglich zu machen, wird von System nur eine einzige Schablone zum Definieren einer bestimmten Geste Klasse erforderlich.

4 Zugrunde liegendes Modell

Wir definieren Gesten so, dass eine Geste ist eine Körpergliedbewegung ist, die durch eine zeitliche Reihe einer festen Anzahl von Parametern vertreten wird. Für eine eingegebene Geste wählt der Erkennungsschritt die beste Übereinstimmung aus einer Menge von Vorlagen aus.

4.1 Kontinuierliche Zustandsmodell

Das Modell kann mit folgendem dynamischen System formuliert werden:

$$\begin{cases} x_k = f_{TR}(x_{k-1}, v_{k-1}) \\ z_k = f_{OB}(x_k, w_k; g) \end{cases} \tag{1}$$

wobei k diskreter Zeitpunkt ist.

- x_k ist ein Vektor, der den Systemzustand und Zustand der Elemente unterschiedlichen Geste Eigenschaften vorstellt
- f_{TR} ist eine Funktion, die die Entwicklung des Systemzustands regelt
- v_k ist eine Rauschsequenz
- f_{OB} ist eine Funktion, die die Beobachtungen z_k erzeugt, je über den Systemzustand x_k und Messrauschsequenz w_k und eine Vorlage Geste g.

Das Problem wird als ein Verfolgungsproblem formuliert.

4.2 Zustandsraummodel

Der Zustand des Systems besteht aus unterschiedlichen Merkmalen und Eigenschaften von Gesten, die über eine lange Zeit geschätzt werden. Der Zustandsraum umfasst die Merkmale, die online bewertet können und folglich als Dauerleistungen während der Interaktion verwendet werden. Die Funktionen werden in jedem Zeitschritt aktualisiert.

4.3 Zustandsübergang

In diesem Modell ist die Zustandsübergangsfunktion f_{TR} linear und durch Matrix A gegeben und als Gauß-Verteilung modelliert:

$$\begin{aligned} p(X_k|X_{k-1}) &= N(X_k|A_{X_{k-1}}, \sum) \\ \sum &= diag(\sigma_1 ... \sigma_D) \end{aligned} \tag{2}$$

Durch Festlegen einer Beziehung zwischen Phase und Geschwindigkeit der Geste wurde eine Einschränkung in Form einer Bewegungsgleichung erster Ordnung gemacht.

$$p_k = p_{k-1} + V_k/T + N(0, \sigma_1) \tag{3}$$

wobei T Lange der Vorlage und σ_1 das erste Element in der Diagonale der$\sum$. Diese Einschränkung kann durch Einstellen der ersten Zeile der Matrix A in $(1\frac{1}{T}0...0)$ berücksichtigt werden. Die andere Bedingungen werden in der ersten Zeile auf Null gesetzt. Die Übergangsparameter spielen eine wichtige Rolle für die Anpassung. Sie regeln die Dynamik der Veränderungsschätzung: die Geschwindigkeit der Annäherung an die genaue Schätzung und die Präzision der Schätzung.

4.4 Beobachtungsfunktion

Die Beobachtungsfunktion wertet die Genauigkeit der Zustandsschätzung gemäß Eingangsbeobachtung und der Vorlage. Parameter der Beobachtungsfunktion legen fest, wie stark die Methode abweicht.

$$S_t(Z_k|f(X_k, g(p_k)), \sum, v) = C(\sum, v)(1 + \frac{d^2(Z_k, f(X_k, g)}{v})^{-\frac{v+K}{2}} \tag{4}$$

wo:

$$C(\sum, v) = \frac{\Gamma(v/2+K/2)}{\Gamma(v/2)} \frac{|\sum|^{-1/2}}{(v\pi)^{K/2}} \tag{5}$$

wobei $f(X_k, g)$ eine Funktion des Templates g und den Zustandswert bei k ist. $f(X_k, g)$ passt sich die erwartete Vorlage Probe g (Stück), da die Phase p_k bei k gegeben ist. Der Abstand d zwischen der angepassten Vorlageprobe und dem eingehenden Beobachtung ist gegeben durch:

$$d(Z_k, f(X_k, g)) = \sqrt{[Z_k - f(X_k, g)]^T \sum^{-1} [Z_k - f(X_k, g)]}. \tag{6}$$

4.5 Inferenz und Implementation

Inferenz ist die Echtzeitschätzung der Zustandswerte. Hier wird die Zustandsprobe aus einer einfacheren Verteilung entnommen und entsprechend ihrer Bedeutung bei der Schätzung der "wahren"Verteilung gewichtet. Bei jedem Schritt

k stellt ein Teilchen x_k^i einen möglichen Wert des Zustandsraums, der von seiner Wahrscheinlichkeit w_k^i gewichtet wird. Der Erwartungswert der Merkmale, zu dem Zeitpunkt k ist:

$\hat{X}_k = \sum_{i=1}^{N_s} w_k^i X_k^i$

wobei N_s die Anzahl der Teilchen bezeichnet. Abgeleitete Merkmalswerte $\hat{X}_k$ bilden den Anpassungsprozess, seit dem Zeitpunkt k. Es werden die Variationswerte beurteilt, die als Zustandsvariablen definiert sind.

4.6 Bearbeitung der Anerkennung

Schließlich wird das Modell erweitert, um Erkennung zu behandeln, indem mehrere Schablonen in Betracht gezogen werden. Der Zustandsraum wird geändert, um die wahrscheinlichste Schablone zusätzlich zu den variierenden Gesteneigenschaften zu schätzen. Betrachtet werden M Vorlagen jeweiliger Länge $L_1 ... L_M$ bezeichnet durch $g^1 ... G^M$. Bei der Initialisierung, wird jedem Zustandspartikel x_k^i eine Geste mit Index zwischen $1...M(m_k bezeichnet)$ zugeordnet, auf der Grundlage einer Anfangsverteilung. Im allgemeinen wird eine gleichmäßige Verteilung gewählt. Dies erweitert die Zustandskonfiguration angewandt auf jedes Partikel wie folgt:

$$X_k^i = \begin{pmatrix} X_k^i(1) \\ . \\ . \\ . \\ X_k^i(D) \\ m_k \end{pmatrix} \in \mathbb{R}^D \times \mathbb{N} \tag{7}$$

Die Übergangswahrscheinlichkeit wird wie folgt angepasst:

$$\begin{aligned} &p(X_k^i | X_(^i k-1) = N(X^i - k | A_(X_(^i k-1)), \sum) \\ &\sum = diag(\phi_1 ... \phi_D 0). \end{aligned} \tag{8}$$

Durch Summieren der Gewichte W_k^i der zugehörigen Gestenindizes der Partikel, ist es einfach, die Wahrscheinlichkeit jeder Geste zu berechnen:
$p(g_k^i | g_k^m) = \sum_{j \in J} w_k^j, \forall l \in [1, M], \forall m \in [1, M], m \neq l$ wo:

$$J = j \in [1, N_s] / X_k^j (D+1) = l. \tag{9}$$

5 Erkennung von realen 2D-Gesten

Um die Gestenauswertung zu evaluieren, wurde die Gestendatenbank von Wobbrock et al.[12] verwendet. Diese enthält Daten von 16 Stiftgesten, die von zehn Teilnehmern in drei verschiedenen Geschwindigkeiten jeweils zehn mal aufgenommen wurden. Für die Versuche wurden pro Geste zufällig aus den Daten

eines Teilnehmers jeweils ein Trainings- und ein anderer Testdatensatz ausgewählt. Insgesamt wurden 4 verschiedene Tests durchgeführt, die jeweils 100 Mal wiederholt wurden:

1. Gleiche Testbedingungen, wie sie von Wobbrock[12] verwendet wurden.
2. Einfluss von geänderten Verteilungsparametern.
3. Verwendung von Trainings- und Testdaten mit unterschiedlichen Geschwindigkeiten.
4. Genauigkeit der Erkennung von Gesten, während sie noch nicht abgeschlossen sind.

Bei allen Tests außer 3. wurden Trainings- und Testdaten mit der gleichen Geschwindigkeit gewählt. Da der von Caramiaux[2] beschriebene Gesture Variance Follower(GVF) auch Veränderungen in den Gesten erkennen und ausgeben soll, ergibt sich für diese Tests ein Zustandsraum x_k, der aus der Phase p_k, der Geschwindigkeit v_k, der Skalierung s_k und dem Drehwinkel α_k besteht.

5.1 Erkennung von Beispielen gleicher Geschwindigkeit

Im ersten Test wurde die durchschnittliche Erkennungsrate sowie die Standartabweichung von GVF mit den erreichten Werten von $1 Recognizer[12], DTW und GF verglichen. Im Vergleich zu dem Gesture Follower(GF) erreicht GVF eine bessere Erkennungsrate ($98,11\%$ gegenüber $95,78\%$), was sich darauf zurückführen lässt, dass GVF sich an Skalierung und Drehung anpasst. Auch im Vergleich mit den beiden Offline-Methoden erreicht GVF leicht bessere Erkennungsraten ($98,11\%$ gegenüber $97,27\%$ bzw. $97,86\%$).

5.2 Einfluss der Verteilungparameter auf die Erkennung

Im zweiten Test wurde der Einfluss der Standartabweichung σ und der Parameter v der Student'schen t-Verteilung auf die Erkennungsrate untersucht. Untersucht wurden dabei für σ in Zehnerschritten Werte von 10 bis 150 und für v die Werte 0.5, 1, 1.5 und ∞ (hier entspricht die Verteilung einer Gaussverteilung).

Die Ergebnisse sind in Abbildung 1 zusammen mit den Erkennungsraten der beiden Offline-Methoden dargestellt.

Hier sind zwei Beobachtungen festzustellen. Erstens, dass die Erkennungsrate die Beste ist für beschränkte σ und v. Zweitens, dass die Verwendung einer Student'schen t-Verteilung anstatt einer Gaussverteilung den Vorteil hat, dass die Erkennungsrate wesentlich weniger von σ abhängt.

5.3 Erkennung von Beispielen verschiedener Geschwindigkeit

Im dritten Test wurden die Erkennungsraten von GVF mit denen des $1 Recognizers verglichen. Diesmal wurden allerdings die Vorlage und die Testgeste aus verschiedenen Geschwindigkeitsgruppen verwendet. Abbildung 2 zeigt die Erkennungsraten der unterschielichen Kombinationsmöglichkeiten. Es wird deutlich, dass insgesamt die durchschnittliche Erkennungsrate beider Algorithmen

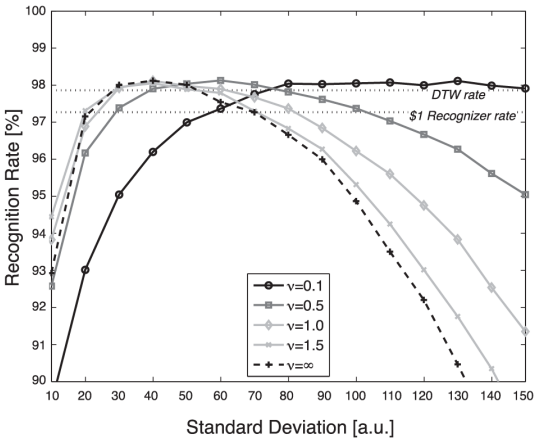

Abb. 1. Einfluss des Parameters v und der Standardabweichung auf die Erkennungsrate

Table I. Results Obtained on a Unistroke Gesture Database Presented by Wobbrock et al. [2007]

	$1 recognizer	DTW	GF	GVF
	offline operated after scaling and rotation estimation		**online** no adaptation of scaling neither rotation	**online** incremental adaptation of scaling and rotation
Mean	97.27 %	97.86 %	95.78 %	**98.11** %
Std	2.38 %	1.76 %	2.06 %	2.35 %

Our model has the following parameterization: $\sigma = 130$, $\nu = 0.1$.

Abb. 2. Erkennungsraten verschiedener Erkennungsmethoden

vergleichbar ist (94,6% gegenüber 94,8%). Ebenso fällt auf, dass die schlechtesten Ergebnisse erreicht werden, wenn Vorlage und Testgeste entgegengesetzte Geschwindigkeiten haben, mit Genaunigkeiten von 88,3% bzw. 85,9%.

5.4 Früherkennung der Gesten

Zuletzt wurde die Erkennung der Gesten untersucht, während sie ausgeführt werden. In Graph Z sind die Früherkennungsraten von GVF und GF in Abhängigkeit vom Fortschritt der Geste dargestellt. Es wird deutlich, dass GVF über den ganzen Messbereich bessere Erkennungsraten erreicht als GF. So erzielt GVF im Schnitt bereits nach 10% der Geste eine Genauigkeit 67% und erreicht bereits nach 40% Fortschritt 90% Genauigkeit.

5.5 Ergebnisse der Versuche mit realen Daten

In den Versuchen hat sich gezeigt, dass GVF in der Erkennung von Gesten gleich gut bis besser ist als andere aktuelle Erkennungsmethoden. Darüber hinaus wurde deutlich, dass die im GVF verwendete Student'sche t-Verteilung gegenüber der im GF verwendeten Gaussverteilung weniger auf eine gute Abschätzung von σ angewiesen ist. Somit ist diese Methode besser in Anwendungsfällen verwendbar, in denen wenig Trainingsdaten zur Verfügung stehen, und somit σ schlecht abgeschätzt werden kann. Eine Verbesserung gegenüber bisher üblichen Methoden stellt der GVF auch dadurch dar, dass bereits während der Gestenausführung Ergebnisse geliefert werden, und die Früherkennung gut genug funktioniert, dass schon nach 10% der Geste Erkennungsgenauigkeiten von über 60% erreicht werden.

6 Abschätzung der Varianz anhand synthetischer Daten

Um die Anpassung an Varianz der Gesten auszuwerten, wurden synthetische Daten verwendet, da menschliche Eingaben keine exakten Werte für die Varianzparameter liefern würden, mit denen die Ergebnisse des GVF verglichen werden könnten. Es wurden zwei Fälle betrachtet: Im ersten Fall wurde nur die Phase und Skalierung der Gesten variiert, im zweiten Fall zusätzlich auch die Rotation. Für die Generierung der Daten wurde eine Viviani-Kurve verwendet:

$$C(t) = \begin{cases} x(t) = a(1 + \cos(t)) \\ y(t) = a\sin(t) \\ z(t) = 2a\sin(t/2) \end{cases} \tag{10}$$

6.1 Auswertung der Phasenabschätzung

Für diesen Fall wurde als Vorlage eine lineare Abstastung der Viviani-Kurve, und als Testdaten eine kubische Abtastung($t \to t^3$) zu der ein gleichmäßig verteiltes Rauschen hinzugefügt wurde, verwendet. Der Zustandsraum ist hier

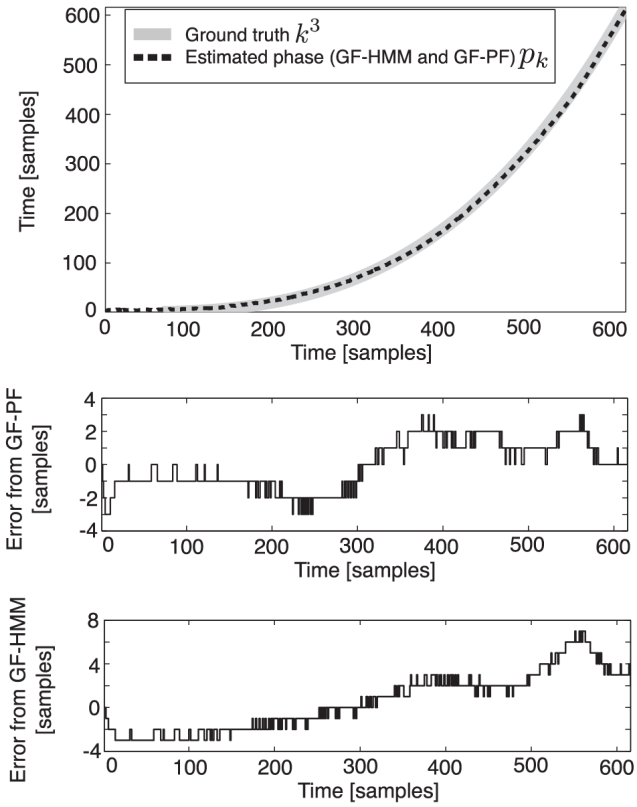
Ground truth k^3
Estimated phase (GF-HMM and GF-PF) p_k
Time [samples]
Time [samples]
Error from GF-PF [samples]
Time [samples]
Error from GF-HMM [samples]
Time [samples]

Abb. 3.

dreidimensional, bestehend aus der Phase $p_k \in [0, 1]$, der Geschwindigkeit $v_k \in \mathbb{R}$ und der Skalierung $s_k \in \mathbb{R}$, wobei v_k und s_k so normalisiert sind, dass ein Wert von 1 jeweils der Geschwindigkeit bzw. Skalierung der Vorlage entspricht. Um einen Vergleich zu GF machen zu können, wurde $v \to \infty$ gewählt.

Im Vergleich erreichten beide Verfahren gute Abschätzungen mit durchschnittlichen Fehlern von 1,3 Abtastungen beim GVF, bzw. 2,3 Abtastungen beim GF.

Weiter wurde der Einfluss von σ auf die Abschätzungen untersucht. Die Ergebnisse sind in Abbildung 3 dargestellt. Für alle getesteten Werte für σ lieferte GVF bessere Abschätzungen als GF, obwohl GF eine genauere Inferenztechnik verwendet. Dies liegt daran, dass GVF ein besseres kontinuierliches Modell verwendet, um die Daten wiederzuspiegeln, sowie an dem Zusammenhang zwischen Geschwindigkeit und Phase, welcher von GF ignoriert wird.

Ebenso stellte sich heraus, dass GVF bessere Abschätzungen in Abhängigkeit von der Stärke des Rauschens liefert, als GF.

6.2 Auswertung der Rotationsabschätzung

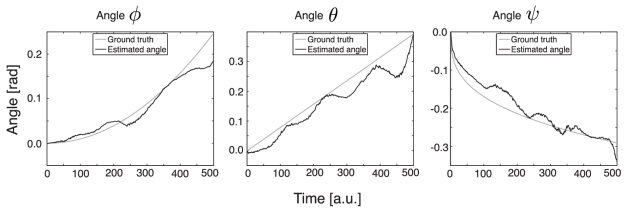

Abb. 4.

Für diesen Versuch wurde die Rotation der Geste mit der Zeit variiert. Die Drehwinkel ϕ, θ und ψ für die Drehung entlang der x, y und z-Achse wurden pro Zeitschritt berechnet durch:

$$\begin{aligned} \phi(t) &= t^2 \\ \theta(t) &= t \\ \psi(t) &= -t^{1/3} \end{aligned} \tag{11}$$

Der Zustandsraum ist hier entsprechend 5-dimensional, bestehend aus p_k, v_k, ϕ_k, θ_k und ψ_k. Es wurden die gleichen Tests durchgeführt wie bei der Phasenabschätzung. Es stellt sich heraus, dass σ wenig Einfluss auf die Genauigkeit der Abschätzung hat, aber dass bei niedrigen Werten für σ die Fehlerrate stärker schwankt als bei großen Werten.

6.3 Ergebnisse der Auswertung synthetischer Daten

In beiden Experimenten wurde deutlich, dass σ wenig Einfluss auf die Abschätzung hat. Daher ist der Algorithmus auch anwendbar, wenn es nur sehr wenig

Trainingsdaten gibt. Ebenso stellte sich heraus, dass die Phasenabschätzung bei festem Rauschen und σ mit einem Durchschnittsfehler von 2,3 Abtastungen sehr genau ist. Zuletzt hat die Stärke des Rauschens einen zu erwartenden Einfluss auf die Abschätzung, wobei sie noch gut genug bleibt, dass der Algorithmus auch bei signifikantem Rauschen verwendet werden kann.

7 Nutzerstudie

7.1 Durchgeführte Experimente

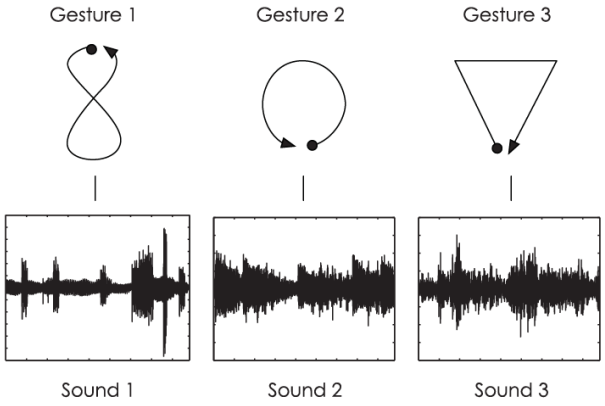

Abb. 5. Die von den Teilnehmern durchgeführten Gesten mit den dazugehörigen Tönen

Es wurde eine Nutzerstudie durchgeführt, um die Einbindung von GVF in reale Anwendungen zu untersuchen. Hierfür wurde eine Anwendung entwickelt, wo eine Gestenausführung die Wiedergabe eines vordefinierten Tons hervorruft, welcher durch Variationen in der Geste verändert wurde. Hierbei wird die Früherkennung des GVF verwendet, um den Ton abzuspielen, sobald die Wahrscheinlichkeit der Erkennung über 50% beträgt. Ab dann werden für den Rest der Geste die Abschätzung der Varianzen für die Manipulation des Tons verwendet. Hierbei sorgt eine schnellere/langsamere Ausführung der Geste für eine schnellere/langsamere Wiedergabe des Tons, größere/kleinere Gesten sorgen für eine lautere/leisere Wiedergabe und eine Drehung der Geste regelt die Cut-Off-Frequenz eines Hochpassfilters. Den Teilnehmern werden 7 Aufgaben gestellt, die alle daraus bestehen, einen bestimmten Ton abzuspielen, und ihn dann zu manipulieren. Hierbei ist zu beachten, dass Aufgabe 1 nur das Abspielen eines Tons, Aufgabe 2-4 eine globale Änderung eines Aspekts, Aufgabe 5 eine kontinuierliche Änderung eines Aspekts und Aufgabe 6 und 7 die globale Änderung zweier Aspekte beinhalten. Die Gesten werden von einem infrarotbasiertem System aufgenommen, welches Fingerbewegungen im freien Raum misst. Die Studie

wurde mit 10 Teilnehmern durchgeführt, die jede der 3 Gesten 3 Mal pro Aufgabe durchführten. Somit ergeben sich 630 Datensätze. Jede aufgenommene Geste sowie der zugehörige wiedergegebene Ton wurden im Nachhinein analysiert, um die Abschätzung des Algorithmus auszuwerten.

Zunächst wurde die Erkennungsrate allgemein untersucht. Insgesamt wurden nur 17 Gesten nicht erkannt, es ergibt sich somit eine Erkennungsgenauigkeit von $97,3\%$. Anschließend wurde die Früherkennung der Gesten untersucht. Hierbei galt eine Geste als erkannt, sobald die Erkennungswarscheinlichkeit größer als 50% ist. Im Durchschnitt wurde eine Geste nach $12,6\%$ der Durchführung erkannt.

7.2 Anpassung einer Eigenschaft

In den Aufgaben 2 bis 4 wurden die Teilnehmer beauftragt, eine Eigenschaft (Lautstärke, Geschwindigkeit und den Hochpassfilter) zu manipulieren. Die Ergebnisse wurden mithilfe eines Student'schen T-Tests mit den Ergebnissen von Aufgabe 1 verglichen.

lauteres Abspielen des Tons

7.3 Beurteilung der Erkennungsgenauigkeit

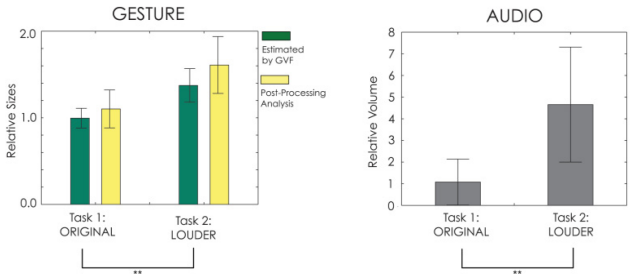

Abb. 6. Ergebnisse von Aufgabe 2 im Vergleich mit 1

In Abbildung 6 wird deutlich, dass die Teilnehmer in der Lage waren, den Ton lauter als normal abzuspielen, da es signifikante Unterschiede der Mittelwerte von Aufgabe 1 und 2 gab. Ebenso zeigt sich, dass die Teilnehmer die Gesten wie gefordert größer ausgeführt haben, was von der Abschätzung auch wiedergespiegelt wurde.

schnelleres Abspielen des Tons Auch hier waren die Teilnehmer in der Lage, die Aufgabe zu erfüllen, da die durchschnittliche Länge eines Tons kürzer war als

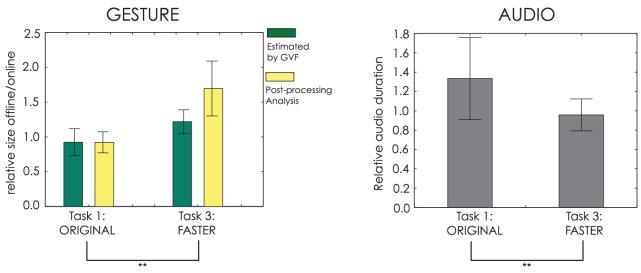

Abb. 7. Ergebnisse von Aufgabe 3 im Vergleich mit 1

bei Aufgabe 1. Es ist aber zu bemerken, dass die Teilnehmer, wenn gebeten den Ton in Originalgeschwindigkeit abzuspielen, ihn etwas langsamer als das Original abspielten. Auch in diesem Versuch erkannte der Algorithmus eine Zunahme der Geschwindigkeit der Geste.

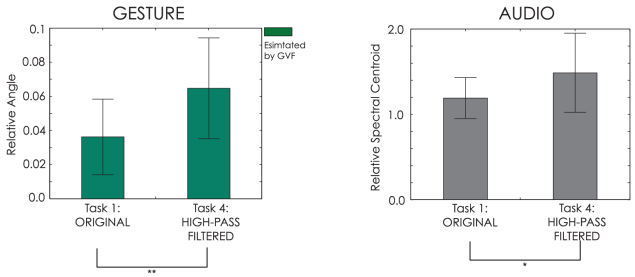

Abb. 8. Ergebnisse von Aufgabe 4 im Vergleich mit 1

Beeinflussung des Hochpassfilters Obwohl mangels Referenzpunkte für diese Aufgabe keine Nachanalyse der Geste möglich war, wurde vom Algorithmus eine Drehung der Geste gemeldet, was sich in der Anwendung des Hochpassfilters deutlich macht.

7.4 kontinuierliche Änderung einer Eigenschaft

In Aufgabe 5 wurde von den Teilnehmern verlangt, den Ton zumächst laut zu spielen und während der Ausführung der Geste leiser zu werden. Wenn man die geschätzten Werte bei 0, 50 und 100% Fortschritt betrachtet, so wird deutlich, dass die Größe der Geste zunächst zunimmt und in der zweiten Hälfte wieder fällt. Somit wird deutlich, dass die Teilnehmer die Aufgabe erfolgreich bewältigt haben.

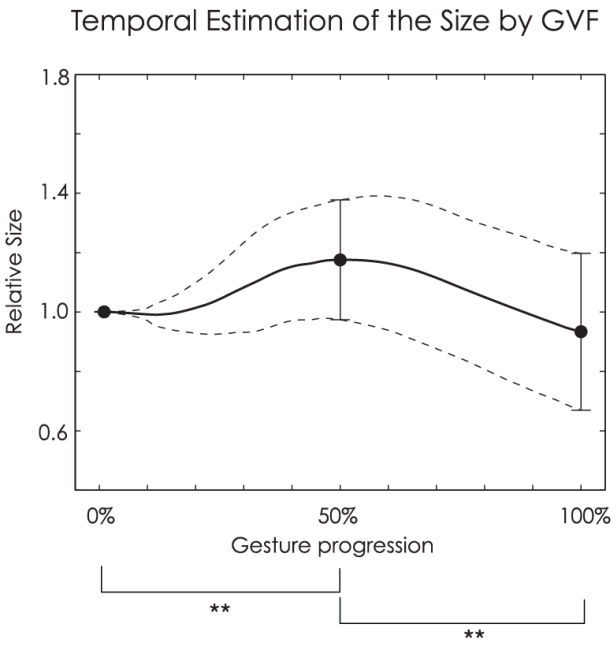

Abb. 9. Geschätzte Geschwindigkeit im Verlauf der Geste

7.5 Änderung zweier Eigenschaften

In den Aufgaben 6 und 7 wurde von den Teilnehmern verlangt, den Ton lauter und langsamer, bzw. leiser mit Hochpassfilter zu spielen. Wie aus den Abbildungen 10 und 11 deutlich wird, ist dies den Teilnehmern gelungen.

7.6 Beobachtungen der Nutzerstudie

Insgesamt haben die Teilnehmer die ihnen gestellten Aufgaben verstanden, und haben es geschafft, die verlangten Ergebnnisse zu erzielen. Die Analyse der wiedergegebenen Töne und abgeschätzten Variationen macht deutlich, dass der Algorithmus die Gesten richtig abschätzt.

8 Fazit

Caramiaux et al.[2] beschreiben einen Algorithmus zur Erkennung von Gesten, der Variationen in Echtzeit mitteilen kann. Da er einen Vorlagenbasierten Ansatz verfolgt, wurde er mit anderen Algorithmen dieser Klasse verglichen. In diesen Tests erreichte er durchweg vergleichbare bis bessere Resultate als die anderen Ansätze. Es wurde kein Vergleich zu Algorithmen durchgeführt, die sich auf große Trainingsdatenmengen stützen, da diese nicht in das projizierte Anwendungsfeld

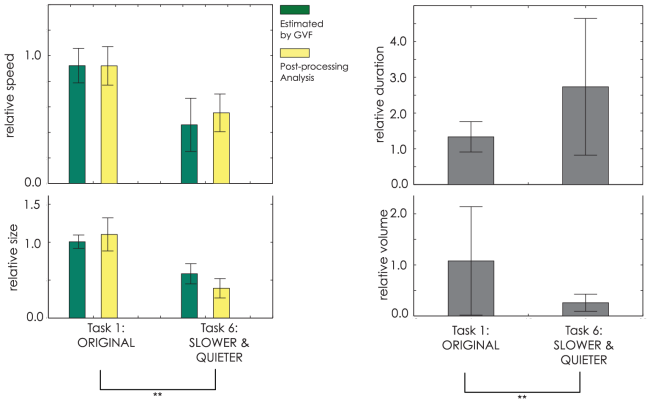

Abb. 10. Ergebnisse von Aufgabe 6 im Vergleich mit 1

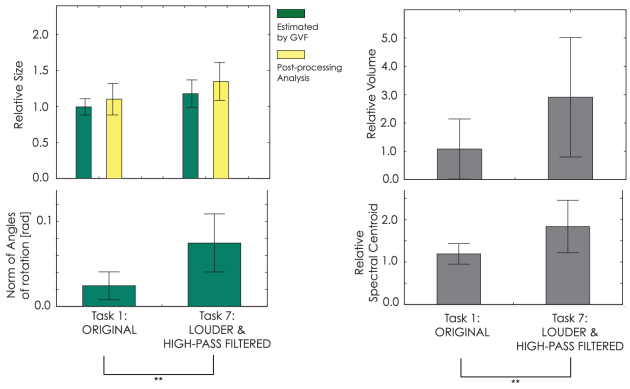

Abb. 11. Ergebnisse von Aufgabe 7 im Vergleich mit 1

fallen. Der Algorithmus ist in der Lage, starke Variationen zwischen Vorlage und ausgeführter Geste zu erkennen, ohne Beispiele für die Variationen zu verlangen. Somit kann eine gute Erkennungsrate mit einzelnen Vorlagen erreicht werden. Ebenso kann der Algorithmus Gesten frühzeitig erkennen und die Variationen feststellen. Insgesamt stellt der Algorithmus also eine Verbesserung gegenüber anderen derzeitigen Methoden dar.

Literatur

1. A.Heloir, N.Courty, S.Gibet, and F.Multon. Temporal alignment of communicative gesture sequences. *Computer Animation and Virtual Worlds 17*, 2006.
2. Baptiste Caramiaux, Nicola Montecchio, Atau Tanaka, and Frédéric Bevilacqua. Adaptive gesture recognition with variation estimation for interactive systems. *ACM Transactions on Interactive Intelligent Systems*, (4), Dezember 2014.
3. D.M.Gavrila and L.S.Davis. Towards 3-d model-based tracking and recognition of human movement: A multi-view approach. *Proceedings of the International Workshop on Automatic Face and Gesture recognition*, 1995.
4. D.Rubine. Specifying gestures by example. *Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques*, 1991.
5. K. Forbes and E. Fiume.
6. J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan. Uwave: Accelerometer-based personalized gesture recognition and its applications. *Pervasive and Mobile Computing 5*, 2009.
7. M.Blackand and A.Jepson. A probabilistic framework formatching tempora ltra-jectories:condensation based recognition of gestures and expressions. *Proceedings of the European Conference on Computer Vision (ECCV'98)*, 1998.
8. D. Merrill and J. A. Paradiso. Personalization, expressivity, and learnability of an implicit mapping strategy for physical interfaces. *Proceedings of the CHI Conference on Human Factors in Computing Systems, Extended Abstracts*, 2006.
9. M.Isard and A.Blake. Condensation conditional density propagation for visual tracking. *International Journal of Computer Vision 29*, 1998.
10. L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 1989.
11. A. D. Wilson and A. F. Bobick. Parametric hidden markov models for gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 21*, 1999.
12. O. Wobbrock, A. D. Wilson, and Y. Li. Gestures without libraries, toolkits or training: A $1 recognizer for user interface prototypes. *Proceedings of the 20th Annual ACM Synopsium on User Interface Software and Technology*, 2007.