

Kurzfassung der Masterthesis: Gestenerkennung mit der SFA

Klassifizierung von beschleunigungsbasierten 3D-Gesten des Wii-Controllers

Kristine Hein, Matrikelnummer: 11048161
Fachhochschule Köln, Campus Gummersbach
Steinmüllerallee 1, 51643 Gummersbach
kristine.hein@gm.fh-koeln.de

Die Grundidee der Arbeit ist es die Slow Feature Analysis (SFA) auf ihre Einsatzmöglichkeiten zur Gestenerkennung zu untersuchen. SFA ist ein Lernalgorithmus, der aus den Neurowissenschaften stammt und aus sich schnell verändernden Signalen, die langsamsten Merkmale herausfiltert [1], [2], [3]. Als Gesten werden in diesem Fall beschleunigungsbasierte dreidimensionale Signale des Wii-Controllers¹ betrachtet. Für die Untersuchungen wurden fünf unterschiedliche Gesten (*circle*, *throw*, *frisbee*, *bowling* und *z*) von zehn Probanden verwendet. Diese Gesten liegen in Form von mehrdimensionalen Zeitreihen vor, die die Messwerte der Beschleunigungssensoren des Wii-Controllers enthalten. Die Gestendaten wurden mit einer Java-Anwendung basierend auf dem WiiGee-Framework [7] aufgezeichnet, mit der die Wiimote über Bluetooth verbunden ist. Für die Untersuchungen wurde die Implementierung des SFA-Toolkits (SFA-TK [5]) verwendet.

Um Gesten wiedererkennen zu können, sollen diese zunächst aus den Zeitreihen herausgefiltert werden. Dazu müssen Anfang und Ende einer Geste bestimmt werden. Viele Ansätze zur Gestenerkennung markieren die Gestenbereiche durch ein zusätzliches Signal, z.B. durch das Drücken eines Buttons während die Geste ausgeführt wird. Es gibt aber auch Ansätze zur automatischen Segmentierung [11], [9], [10], um Geste- und Nicht-Gesten-Sequenzen zu trennen. Es wurden verschiedene Ansätze zur Segmentierung untersucht und gegenübergestellt: Eine regelbasierte Segmentierung über eine Hysteresefunktion, eine Segmentierung über die durchschnittliche Bewegungsänderung [9] und drei Varianten zur Segmentierung mit SFA. Für alle Verfahren mussten pro Geste optimale Parameter, wie Grenzwerte und Mindestlängen, ermittelt werden. Zur Segmentierung mit der SFA wurde zum einen der Beschleunigungsbetrag als Eingangssignal für eine einfache SFA zweiten Grades verwendet, zum anderen die ursprünglichen Beschleunigungswerte. Dabei hat sich herausgestellt, dass die Resultate beider Untersuchungen ähnlich waren, da bei Verwendung der rohen Beschleunigungswerten die SFA indirekt ebenfalls den Beschleunigungsbetrag findet. Dies kann mit dem Expansionsschritt der SFA begründet werden. Eine weitere Untersuchung zur Segmentierung mit kaskadierter SFA, um nicht nur Monome zweiten Grades zu verwenden, "glättet" die Ergebnisse etwas, brachte bei Verwendung des langsamsten Ausgangssignal der ersten SFA als neues Eingangssignal keine verbesserten Ergebnisse. In allen drei Varianten gruppieren sich die Ruhelagewerte um eine "Grundlinie" die in der Nähe des Medians des gesamten Ausgangssignals liegt. Zur Segmentierung wurden deswegen einfache Grenzen um den Medianwert gelegt und Mindestlängen für Gesten- und Nicht-Gesten-Segmente festgelegt, ähnlich wie bei dem regelbasierten Verfahren. Gemessen wurden die Verfahren anhand ihrer Abweichung zu einem Button-Gestenindikator (einer Gestenmarkierung, die durch Drücken des Buttons bei der Gestenaufzeichnung erhoben wurde). Da dieser nur die subjektiven Gestengrenzen der jeweiligen Person widerspiegelt und auch aus motorischen Gründen nicht exakt sein muss, sollten in weiteren Untersuchungen zum einen der Gesamtprozess der Gestenerkennung bestehend aus Segmentierung und anschließender Klassifizierung näher betrachtet werden und zum anderen dahingehend optimale gesten- und personenübergreifende Parameter gefunden werden, z.B. anhand von Metaregeln.

Sind die Gestensegmente bekannt, so sollen diese einer definierten Gestenklasse zugeordnet werden können. Eine solche Klassifizierung von Gesten ist bereits mit unterschiedlichen Ansätzen untersucht worden, häufig basierend auf dem Hidden Markov Modell (HMM), welches ein gängiges Verfahren ist, das oft im Bereich der Zeitreihenklassifikation eingesetzt wird [6], [9], [10]. Ebenfalls zum Vergleich wurde Random Forest (RF) verwendet; ein sehr robustes

¹ von Nintendo

Klassifizierungsverfahren, bei dem während des Lernprozesses eine Menge von Entscheidungsbäumen generiert wird. Anders als diese beiden Verfahren, "findet" die SFA die entscheidenden Merkmale aus den Daten selbst. Sie wurde zur Klassifizierung bereits im Bereich der Ziffernerkennung erfolgreich eingesetzt [4]. Im Bereich der Gestenerkennung hat die SFA im Vergleich zu anderen gängigen Verfahren, wie RF, HMM oder simplen Gaußklassifizierern vergleichbare und teilweise sogar wesentlich bessere Ergebnisse erzielen können, wie in dieser Arbeit gezeigt wurde. Während RF auch mit wenigen Trainingsdaten gute Ergebnisse erzielt, kommt es hier bei der SFA zu Overfitting-Problemen. Diese konnten aber durch das Ergänzen der Trainingsdaten durch zusätzliche künstlich erzeugte Daten, dem sogenannten Parametric Bootstrap, gelöst werden. Bei steigender Anzahl der künstlichen Daten führte für eine personenabhängige Klassifizierung das Verrauschen der Daten (gaussian noise) am schnellsten zu den besten Ergebnissen und erreichte mit einer Fehlerrate von durchschnittlich 2.5% auf den Testdaten die besten Ergebnisse der gesamten Untersuchung. Mit der SFA-Klassifizierung ließen sich ebenfalls gute Ergebnisse bei einer personenübergreifenden Erkennung erzielen, sogar wenn nur wenige Originaldaten pro Person verwendet wurden. In realen Umgebungen besteht häufig die Notwendigkeit auch Gesten neuer Personen zu erkennen, deren Daten zuvor nicht zum Trainieren des Systems vorhanden waren. Auch hierbei erzielt die Klassifizierung mit SFA vergleichbare Ergebnisse mit RF und schnitt damit wesentlich besser ab als das gängige HMM. In diesem Fall zeigte ein angepasstes PB-Verfahren welches Rotationen der Gestendaten berechnet bei einer größeren Anzahl zusätzlicher Daten die besten Ergebnisse.

In dieser Arbeit wurde gezeigt, dass sich die SFA zur Gestenerkennung einsetzen lässt, obwohl sie nicht speziell auf den Bereich der Gestenerkennung ausgerichtet ist; bei zu wenigen Trainingsdaten unter Einsatz von Parametric Bootstrap. Um in einem dynamischen Gesamtprozess auch eine automatische Segmentierung zu realisieren, sind zukünftig allerdings weitere Untersuchungen notwendig, da hier noch nicht der optimale Weg gefunden wurde.

Literatur

- [1] Laurenz Wiskott. Learning Invariance Manifolds. *Proc. of the 5th Joint Symposium on Neural Computation, San Diego, CA*, May 1998
- [2] Laurenz Wiskott and Terrence Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, 14(4), Apr 2002
- [3] Laurenz Wiskott. Estimating Driving Forces of Nonstationary Time Series with Slow Feature Analysis. *Arxiv preprint cond-mat/0312317*, Dec 2003
- [4] Pietro Berkes. Pattern Recognition with Slow Feature Analysis. *Cognitive Sciences EPrint Archive (CogPrints)*, 4104, 2005
- [5] Wolfgang Konen. On the numeric stability of the SFA implementation sfa-tk. *Arxiv preprint arXiv:0912.1064*, Dec 2009
- [6] Thomas Schlömer, Benjamin Poppinga, Niels Henze, and Susanne Boll. Gesture recognition with a Wii controller. *TEI '08: Proceedings of the 2nd international conference on Tangible and embedded interaction*, Feb 2008
- [7] Benjamin Poppinga and Thomas Schlömer. A Java-based gesture recognition library for the Wii remote. *Carl von Ossietzky Universität Oldenburg*, Aug 2007, <http://www.wiigee.org>
- [8] Patrick Koch, Wolfgang Konen, and Kristine Hein. Gesture Recognition on Few Training Data using Slow Feature Analysis and Parametric Bootstrap. *2010 International Joint Conference on Neural Networks*, 2010
- [9] Zoltán Prekopcsák. Accelerometer Based Real-Time Gesture Recognition. *POSTER 2008*, Prague, May 2008
- [10] Frank G. Hofmann, Peter Heyer, and Günther Hommel. Velocity profile based recognition of dynamic gestures with discrete hidden Markov models. *Gesture and Sign Language in Human-Computer Interaction: International Gesture Workshop, Bielefeld, Germany, September 1997. Proceedings*, Lecture Notes in Computer Science(Volume 1371/1998), Oct 1998
- [11] AiLive Inc. AiLive LiveMove Pro, Oct 2006, <http://www.aillive.net/liveMovePro.html>, Stand: 05.05.2010