Realisation einer FPGA-basierten psychometrischen Authentifizierung mittels Touchscreen

Erik Raik Engelhardt

Hochschule für angewandte Wissenschaften – Hamburg

erik.engelhardt@haw-hamburg.de

17. Mai 2019

Einleitung

Ziel dieser Arbeit

Kann man einen Nutzer anhand des Eingabeverhaltens an einem Touchscreen erkennen?

Motivation

- Eine Pin oder ein Passwort kann leicht ausspioniert und kopiert werden
- Ein biometrisches Merkmal kann kaum bis gar nicht ersetzt werden

Ziel dieser Arbeit

Kann man einen Nutzer anhand des Eingabeverhaltens an einem Touchscreen erkennen?

- Eingabeverhalten bei der Eingabe eines Pin-Codes auf einem Touchscreen
  - Einbeziehung der Position, des Druck und des Zeitverhalten
- Evaluierung verschiedener Klassifizierungsalgorithmen
- Implementierung auf einem FPGA
- Testen der Implementierung in einem Feldtest

Grundlagen

Grundsätzlicher Aufbau

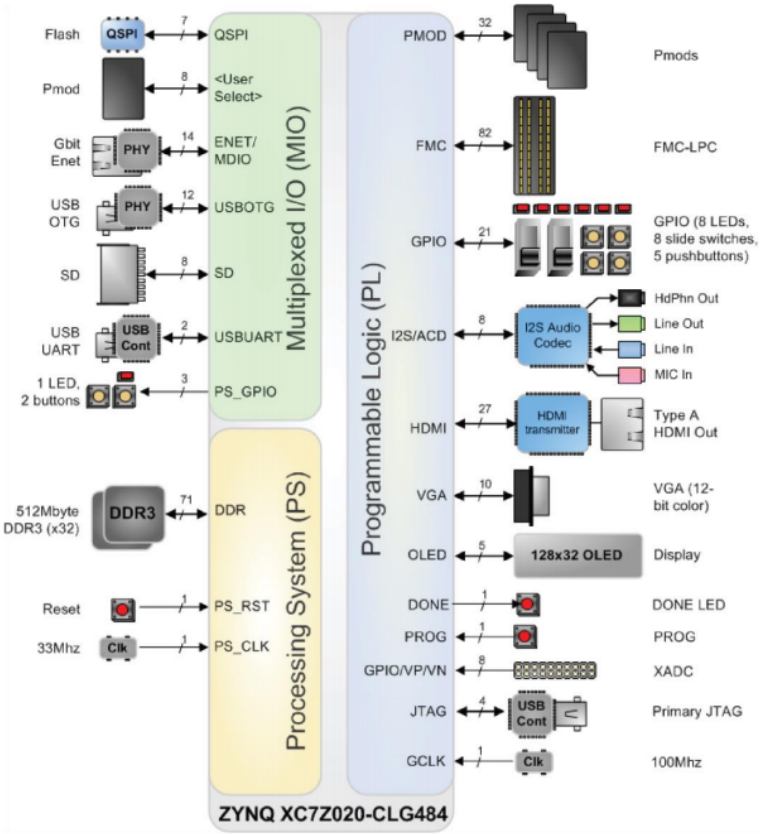

Abbildung: Zedboard Komponentenübersicht [AVNET]

Erik Raik
Engelhardt

Support Vector Machine

- Trennen der Datensätzen durch Hyperebene
- Maximieren des Abstands der Datenpunkte zur Hyperebene

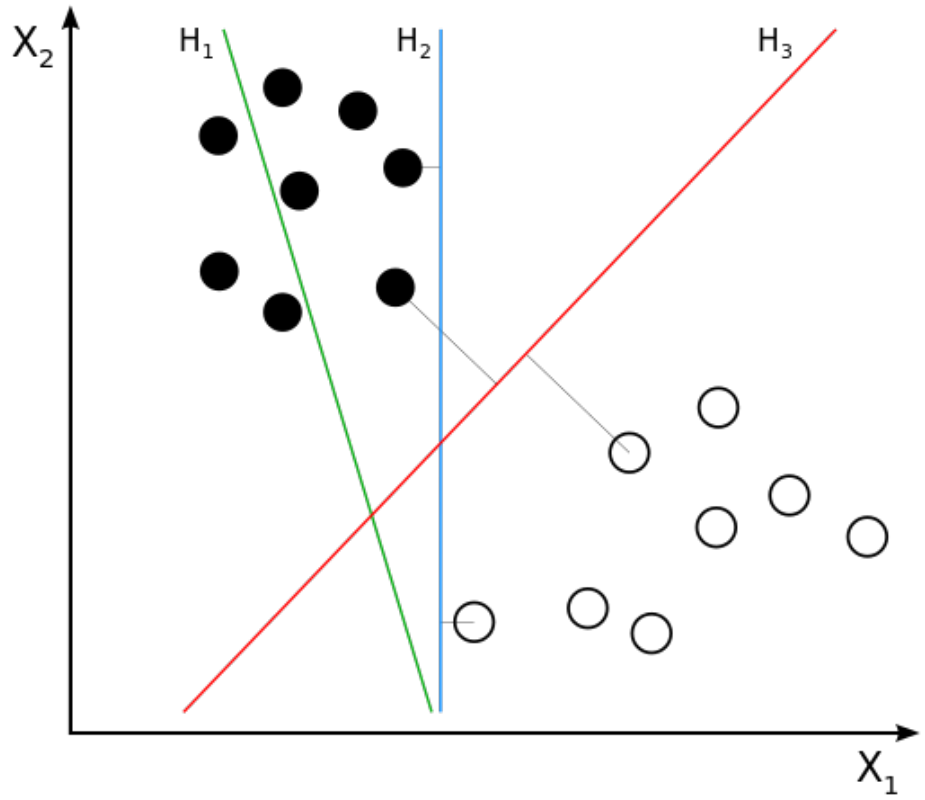

Abbildung: Beispiele für Hyperebenen einer SVMs [Yim]

Erik Raik
Engelhardt

Implementierung

Touchscreen-Anschlussboard

- Erkennen und Melden einer Berührung
- Messen der Berührungsposition
- Weiterleitung der Daten an die PL des Zedboards

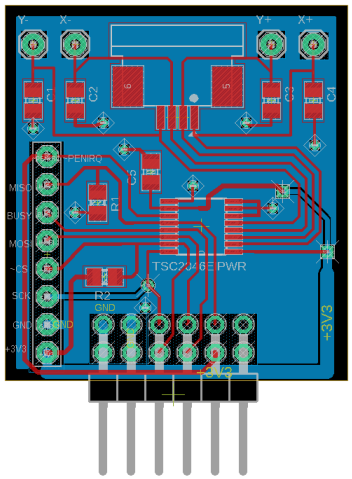

Abbildung: Platinenlayout für das Touchscreen-Anschlussboard. Obere Lage in Rot, untere in Blau. Verbindungen der Lagen in Grün.

Erik Raik
Engelhardt

Touchscreen-IP-Core

- Auslesen der Positionsdaten
- Erfassen von Timingdaten
- Bereitstellung dieser Daten an das PS

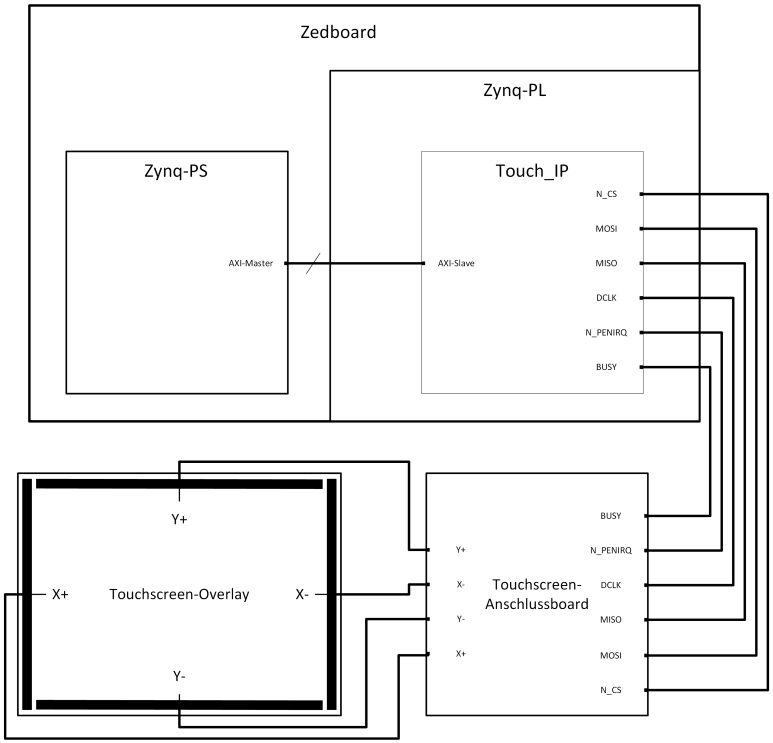

Abbildung: Grundsätzlicher Aufbau des Touchsystems

Erik Raik
Engelhardt

HDMI-IP-Core

- Generierung der Bild- und Timingdaten für das HDMI-Display
- Weiterleitung dieser Daten an den HDMI-Transceiver
- Darstellung von Inhalten aufgrund von Anweisungen aus dem PS

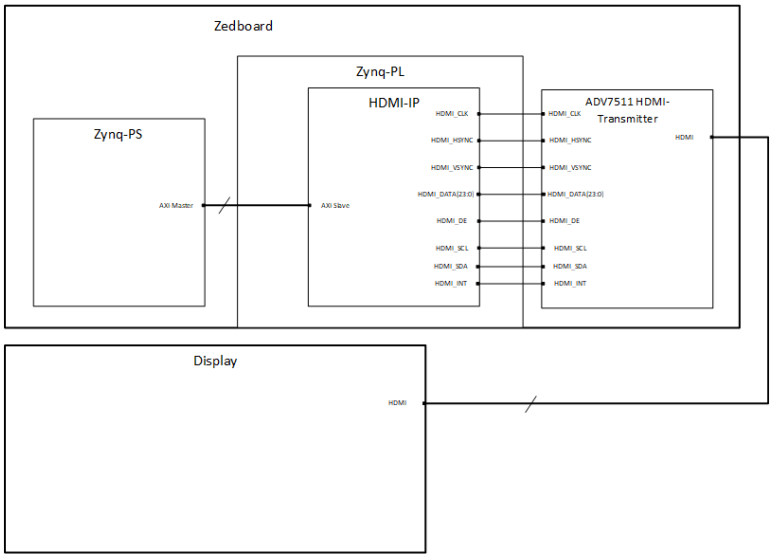

Abbildung: Grundsätzlicher Aufbau des Display Systems

Erik Raik Engelhardt

Aufgaben des Processing Systems

- Steuerung der Bildinhalte und damit die Kommunikation mit dem HDMI-IP-Core
- Auslesen der Messdaten vom Touch-IP-Core
- Filtern und Transformieren der Messdaten
- Speichern der Messdaten auf einer SD-Karte
- Ausführen des Authentifikationsalgorithmus
- Menüführung über das OLED-Display und die Taster des Zedboards

Datenerhebung

- Erfassung des Eingabeverhaltens von 16 Personen für 3 verschiedene Pin-Codes
- Insgesamt 948 Eingaben über 11 Tage erfasst
- Testpersonen wurden über betrachtete Merkmale in Kenntniss gesetzt

Erik Raik
Engelhardt

Trainieren der Klassifikatoren in Python

- Training der Klassifikatoren unter Verwendung der Bibliothek SciPy-Bibliothek [Jones et al. [2001–]]
- Export der Modelle unter Verwendung der Sklearn-Porter Bibliothek [Morawiec]

	DT		RF		KNN		NB		SVM	
A	R	P	R	P	R	P	R	P	R	P
0	0.66	0.71	0.58	0.89	0.73	0.86	0.86	0.49	0.87	0.71
1	0.69	0.73	0.57	0.94	0.76	0.86	0.95	0.45	0.91	0.74
2	0.72	0.77	0.57	0.93	0.85	0.88	0.88	0.62	0.89	0.87
	0.69	0.74	0.57	0.92	0.78	0.86	0.9	0.52	0.89	0.77

Tabelle: Durchschnittliche Recall (R) und Precision (P) über alle Nutzer pro Account und Algorithmus. Durchschnittswerte über alle Accounts in der letzten Zeile.

Erik Raik Engelhardt

Recall und Precision

Recall:
In wie viel Prozent der Fälle wurde der Nutzer richtig erkannt?

$$R = \frac{TP}{TP + FN}$$

Precision:
In wie viel Prozent der Fälle in denen der Algorithmus gedacht hat es handle sich um den Nutzer war dies auch wirklich der Fall?

$$P = \frac{TP}{TP + FP}$$

Trainieren der Klassifikatoren in Python

- Training der Klassifikatoren unter Verwendung der Bibliothek SciPy-Bibliothek [Jones et al. [2001–]]
- Export der Modelle unter Verwendung der Sklearn-Porter Bibliothek [Morawiec]

	DT		RF		KNN		NB		SVM	
A	R	P	R	P	R	P	R	P	R	P
0	0.66	0.71	0.58	0.89	0.73	0.86	0.86	0.49	0.87	0.71
1	0.69	0.73	0.57	0.94	0.76	0.86	0.95	0.45	0.91	0.74
2	0.72	0.77	0.57	0.93	0.85	0.88	0.88	0.62	0.89	0.87
	0.69	0.74	0.57	0.92	0.78	0.86	0.9	0.52	0.89	0.77

Tabelle: Durchschnittliche Recall (R) und Precision (P) über alle Nutzer pro Account und Algorithmus. Durchschnittswerte über alle Accounts in der letzten Zeile.

Erik Raik Engelhardt

Feldtest

Durchführung

- Implementierung der SVM für Account 0, Nutzer 1 auf dem Zedboard
- 13 Testpersonen
- 20 Versuche pro Person
- Aufklärung der Testpersonen über Authentifikationsalgorithmus
- Ermöglichen des Beobachtens der Eingabe nach den ersten 10 Versuchen

Erik Raik Engelhardt

Ergebnisse

Nutzer	Erfolgreiche Versuche (absolut)	Erfolgreiche Versuche (prozentual)	Erster Erfolgreicher Versuch
1	19	0.95	1
2	2	0.10	16
3	0	0.00	-
10	1	0.05	12
14	1	0.05	14
17	4	0.20	12
18	0	0.00	-
19	0	0.00	-
20	0	0.00	-
21	1	0.05	14
22	5	0.25	5
23	2	0.10	19
24	1	0.05	13

Tabelle: Ergebnisse der Validierung der SVM (trainiert auf Nutzer 1, Account 0) im Feldtest

Erik Raik
Engelhardt

Ergebnisse

- Recall: 95%
- Precision: 53% (Hochrechnung für gleiche Anzahl an Eingaben für Nutzer und Angreifer: 93%)
- Flase Rejection Rate (FRR): 5%
- False Acceptance Rate (FAR): 7%

Fazit

Bewertung des Verfahrens

- Im Zusammenspiel mit einer klassischen Pineingabe durchaus als Merkmal für eine Authentifizierung geeignet
- Resistent gegenüber „Shoulder Surfing“
- Leichter zu ersetzten als ein biometrisches Merkmal
- Unterschiedliche Performance für verschiedene Nutzer
  - Touchscreen nicht optimal
  - Unterschiedlich viel Erfahrung im Umgang mit Touchscreens
  - Unterschiedlich viele Testdaten

Ausblick

Es gibt noch viele Möglichkeiten die Performance des Verfahrens zu verbessern:

- Umfangreicherer Feldtest unter diverseren Testbedingungen
- Optimieren der Klassifikatoren
- Verwenden verschiedener Klassifikatoren und Abstimmung über ein Mehrheitsvotum
- Verwendung eines zuverlässigeren Eingabegerätes

Literatur- und Quellenverzeichnis I

AVNET. Zedboard getting started guide. URL http://zedboard.org/sites/default/files/documentations/GS-AES-Z7EV-7Z020-G-V7-1.pdf. Zugriff: 12.01.2019.

Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. URL http://www.scipy.org/. Zugriff: 19.04.2019.

Darius Morawiec. sklearn-porter – transpile trained scikit-learn estimators to c, java, javascript and others. URL https://github.com/nok/sklearn-porter. Zugriff: 20.04.2019.

Annie Yim. Essential classification algorithms with explanation. URL https://www.kaggle.com/anniepyim/essential-classification-algorithms-explained. Zugriff: 17.03.2019.