

Titel der Ausarbeitung bzw. des Themas

Proseminar-Ausarbeitung von

thomas

An der Fakultät für Informatik
Institut für Visualisierung und Datenanalyse,
Lehrstuhl für Computergrafik

17. Mai 2018

Inhaltsverzeichnis

1	Einleitung	1
1.1	Zielsetzung der Arbeit	1
1.2	Gliederung der Arbeit	1
2	Grundlagen	3
2.1	Taktile Geräte	3
2.2	Genetischer Algorithmus	3
2.2.1	Evolution in der Biologie	3
2.2.2	Selektion	4
2.2.3	Variation	4
2.2.4	Gendrift	4
2.2.5	Allgemeiner Vorgang eines Evolutionären Algorithmus	4
2.3	Verwandte Arbeiten	6
2.3.1	Taptic Engine	6
2.3.2	Personalisierte Vibration	6
2.3.3	Personalisierte Smartwatch	8
2.3.4	Fazit	8
2.3.5	Design of a Wearable Tactile Displays	8
3	Analyse	11
3.1	Anforderungen	11
3.2	Existierende Lösungsansätze	13
3.3	Weiterer Abschnitt	14
3.4	Zusammenfassung	14
4	Entwurf	15
4.1	Programm Ablauf	15
4.2	Ausführung des Programms	15
4.2.1	Signal	15
4.3	Evolutionärer Algorithmus	16
4.4	Evolutionärer Algorithmus	16
4.4.1	Eingabe für den Algorithmus	16
4.4.2	Muster	19
4.5	Zusammenfassung	20
5	Implementierung	21
5.1	Signal	21
5.2	Muster	21
5.3	Evolutionärer Algorithmus	21
6	Evolutionärer Algorithmus	23
6.1	DNA	23
6.2	Population	23

6.3	Generationen	23
6.4	Mutation	23
6.5	Studiendesign	23
6.5.1	Studiendesign	23
7	Evaluierung	25
7.1	Initialisierung der Grenzen	25
7.2	Abschnitt 2	25
7.3	Zusammenfassung	25
8	Zusammenfassung und Ausblick	27
9	Zusammenfassung und Ausblick	29
	Literaturverzeichnis	31

1. Einleitung

Hinweis: In die Einleitung gehört die Motivation und Einleitung in die Problemstellung. Die Problemstellung kann in der Analyse noch detaillierter beschrieben werden.

1.1 Zielsetzung der Arbeit

Was ist die Aufgabe der Arbeit? ...

Mit dieser Bachelorarbeit verfolge ich das Ziel personalisierte Vibrationssignale zu erstellen. Hierbei werden drei verschiedene Vibrationssignale für einen Nutzer so angepasst, dass die Werte speziell für den Benutzer bestimmt werden. Zur Bestimmung der eines Vibrationssignals wird ein Evolutionärer Algorithmus verwendet. Nachdem der Evolutionäre Algorithmus passende Wert gefunden hat, wird überprüft, wie gut die personalisierten Vibrationssignale im Vergleich zu vorgegebenen Werten erkannt werden.

//Die Aufgabe besteht darin, für ein Individuum mit einem Programm drei verschiedene Vibrationssignale für Ihn so anzupassen, dass die Werte von Ihm besser erkannt werden als vorgegebene Werte. Dabei wird zur Bestimmung eines Vibrationssignals ein Evolutionärer Algorithmus verwendet. Nachdem dieser passende Wert gefunden worden ist, wird überprüft, wie gut im Vergleich zu Vorgegebenen Werten die Daten erkannt werden.

Somit wollte ich herausfinden, ob es möglich ist für jedes Individuum eine eigene passende Länge und Stärke von Vibrationssignalen zu erstellen, so dass die Kombination der Signale noch erkannt werden.

Die Hypothese die ich mit dieser Bachelorarbeit beantworten möchte ist, ob man mittels personalisierten Vibrationen eine Folge von Vibrationen besser unterscheiden als vorgegebene Vibrationen. Um dies beantworten zu können verwende ich Wearable zur Darstellung der Vibrationssignale und einen Genetischen Algorithmus um die personalisierten Vibrationen zu ermitteln.

1.2 Gliederung der Arbeit

Was enthalten die weiteren Kapitel? ...

Im Verlauf dieser Bachelorarbeit erläutere ich erst allgemein, was die einzelnen Bestandteile des Evolutionären Algorithmus sind und wie ich den an mein Problem angepasst habe.

Sowie das Problem ansatzweise heutzutage umgesetzt wurde und wie ich an das Problem heranging und wie ich es Implementiert haben.

// auskommentieren In meiner Bachelorarbeit werde ich erst einmal erläutern, was aktuell in den Smartphones und in der Forschung benutzt wird. Im Folgenden werde ich erläutern, was ein Genetischer Algorithmus ist und wie ich diesen nutze genutzt habe.

2. Grundlagen

Die Grundlagen müssen soweit beschrieben werden, dass ein Leser das Problem und die Problemlösung versteht. Um nicht zuviel zu beschreiben, kann man das auch erst gegen Ende der Arbeit schreiben.

Bla fasel...

2.1 Taktile Geräte

Ein Taktiler Gerät ist ein Gerät, das Informationen an einen Menschen mitteilen möchte, dies geschieht durch die Wahrnehmung der Haut. Ein Taktiler Gerät ist ein Gerät, das Informationen durch die Wahrnehmung des Menschen mittels Körperkontakt darstellt. [GOS01]

Taktile Geräte werden heutzutage öfter verwendet, als man es eigentlich wahrnimmt. Ein einfaches Beispiel ist das Handy. Eine Person hat sein eigenes Handy in der Hosentasche. Bei einer eingehenden Nachricht, muss der Benutzer mitgeteilt werden, dass eine Nachricht empfangen wurde. Normalerweise geschieht das beim Abspielen des Klingeltons. Falls man beschäftigt ist und nicht durch ein lautes Klingeln gestört werden will, stellt man den Ton ab. Um dennoch darauf aufmerksam zu machen, dass eine Nachricht eingetroffen ist, nutzt man die Vibration des Handys.

Das Handy ist nur eins von vielen Beispielen, was man über den Alltag noch für Taktile Geräte verwendet.

2.2 Genetischer Algorithmus

Unter dem Begriff der Evolutionären Algorithmen (EA) versteht man eine Ansammlung von Techniken und Methoden, die für Optimierungsprobleme eine Lösung finden, die das Problem näherungsweise löst. [?] //Man hat sich bei dem EA an der Biologischen Evolution von Darwin inspirieren lassen. [Sel03]

2.2.1 Evolution in der Biologie

Im 19. Jahrhundert hat sich Darwin mit den Gedanken über die Evolution von Lebewesen über mehrere Generationen gemacht. Dabei kam er auf das Prinzip "Survival of the fittest". Mit dem Prinzip hat Charles Darwin die Entstehung neuer Arten beschrieben. Dabei

haben die Arten überlebt, die sich am besten für die Umgebung angepasst haben. Die Arten, die sich an die neue Situation nicht angepasst hatten sind nach ein paar Generationen zur Minderheit geworden. [?]

In der Biologie ist jeder lebende Organismus ein **Individuum**. Jedes Individuum besitzt Erbinformationen in der Form von Chromosomen. Die Erbinformationen werden auch **Gene** oder **DNA** genannt. Eine Gruppe von Individuen wird als **Population** bezeichnet.

Bei dem EA hat man sich das Verhalten des in der Biologie angeschaut, wie dort die Arten über Generationen hinweg überleben und versucht dieses zu adaptieren. [Fli]

2.2.2 Selektion

Eine **Selektion** ist eine Auswahl von Individuen einer Population. Durch Kombination der Gene der ausgewählten Individuen, werden neue Individuen für die nächste Generation erzeugt. Diese Kombination wird **Rekombination** genannt. Dabei gibt es verschiedene Selektionsstrategien. Man versucht die Individuen zu finden, die eine bestmögliche Lösung für ein Problem liefern. [?, Fli]

2.2.3 Variation

Für eine Population sollte man zu Beginn eine große **Variation** von Individuen mit unterschiedlichen Genen besitzen. Nehme man anhand einem Beispiel an, dass man mithilfe einer Evolution eine neue Art von Süßigkeiten entwickeln möchte. Dabei sollten die Form und Farbe neu bestimmt werden. Wenn man jetzt eine Start-Population von Individuen habe, die alle die gleichen Gene besessen würde, so würde man anhand zwei Selektierten Individuen keine Änderung der nächsten Generation erkennen, da die Nachkommen auch alle die gleichen Gene besitzen würden, falls keine Mutation auftreten würde. Damit dieses Verhalten nicht auftritt, versucht man zu Beginn eine große Variation an Individuen zu erzeugen und diese als Anfangs Population für einen Evolutionären Algorithmus zu nutzen. Mittels Rekombination und Mutation wird dabei ein neues Individuum für die nächste Generation erzeugt, dass neue Gene vorweist. Mittels der Rekombination werden verschiedene Gene bei der Fortpflanzung der nächsten Generation vermischt und somit neue Gene für erzeugt. Die Mutation ist eine Veränderung der Gene eines Individuums, die meistens durch Umwelteinflüsse ausgelöst werden. [Fli]

2.2.4 Gendrift

Unter **Gendrift** versteht man eine zufällige Veränderung der Genhäufigkeit in einer Population, innerhalb einer Evolution. Ein Gendrift tritt bei kleineren Populationen häufiger auf. [?]

2.2.5 Allgemeiner Vorgang eines Evolutionären Algorithmus

Der EA besitzt im allgemeinen immer die gleichen Komponenten. Diese Komponenten werden im folgenden erläutert. Dieses Wissen stammt aus den folgenden Quellen [Shi12, Fli, ?]

Initialisierung

In der Initialisierung erzeugt man sich eine Population von Individuen, die eine zahlreiche Variation von Genen besitzen. Für die Anfangs-Population nimmt man in der Praxis zufällige Individuen oder auch die besten bekanntesten Lösungskandidaten.

Bewertung der Individuen

Bevor man eine Evolution für eine Population erzeugen kann, benötigt man Attribute anhand derer man die nächste Generation berechnet. Mit sogenannten **Fitnessfunktion** berechnet anhand der Attribute eines jeden Individuums einen **Fitnesswert**. Der Fitnesswert ist entscheidend für die Selektion.

Selektion

Die Selektion wählt zwei Individuen als Eltern aus, diese werden anhand der Fitnesswert der Individuen aus einer Population bestimmt. Es werden die Individuen bevorzugt, die einen hohen Fitnesswert aufweisen. Diese ermittelten Eltern werden für die Rekombination und Mutation weiter verarbeitet.

Rekombination

Die Eltern aus der Selektion werden in diesem Schritt einen Nachfahren für die nächste Generation erzeugen. Dabei werden die jeweiligen Gene der Eltern kombiniert und an die nächste Generation vererbt. Die Kombinationsmöglichkeiten hängen davon ab, wie die Gene repräsentiert sind. Die neuen Individuen werden in die Population für die nächste Generation hinzugefügt.

Mutation

Nach der Rekombination besteht eine Chance, dass die Gene der nächste Generation mutieren können.

Terminierung

Man führt für eine Generation die Selektion, Rekombination und Mutation so oft aus, bis man die gleiche Anzahl an Individuen für die nächste Generation erzeugt hat. Der Vorgang der Bewertung, Selektion, Rekombination und Mutation muss für jede neue Generation durchgeführt werden. Die neu erzeugte Population wird als Eingabe für die neue Generation verwendet. Dabei wird der Algorithmus so lange ausgeführt bis eine hinreichende Abbruchbedingung erreicht worden ist. Je nach Definition kann nach einer festen Anzahl an Generationen oder erst nachdem es keine signifikanten Änderungen der Generationen gibt den EA terminieren. Das dabei erzeugte Ergebnis ist eine näherungsweise Lösung für das Problem.

Wiederholung durch neuer Generation

We played the participant 10 equally distributed signals and asked him which signal type he detected. The participant could replay a signal again. The answer that he could pick for a signal is SHORT, MIDDLE or LONG. Before the participant should answer the questions we played all signals on the wristband.

To achieve that the user does not to do random selections, we proofed a few cases.

We defined, that the shortest Signal (here 100ms) should be a SHORT Signal and the longest Signal (here 1000ms) should be a LONG Signal. The user should at least use 2 times one signal type as an answer. The borders of the Signal types should be rising. If one of these cases wouldn't be true, than the participant should do it again. That happened just by 15,625 % of all participants (5 of 32), that they should repeat it a second time.

To define the borders we picked the first and last SHORT as their border. The signal after the last SHORT would be a MIDDLE. This MIDDLE Signal is the first border and the last border is the last MIDDLE. The same for the LONG border.

We got three borders, once for each signal type. The SHORT border is defined as the first and last answer of SHORT. The signal followed of the last SHORT is a the typical a MIDDLE answer. It is taken as the first border of MIDDLE and the last of that type is the last border. The same for the LONG border.

The easiest way would be if the participant select SHORT for the first elements followed with just MIDDLE and just LONG answers. You may think that this is just a rare example, but actually 75 % of the participants had a result like this. In the other 25 % the user just swapped up to 1 elements.

2.3 Verwandte Arbeiten

Hier kommt „Related Work“ rein. Eine Literaturrecherche sollte so vollständig wie möglich sein, relevante Ansätze müssen beschrieben werden und es sollte deutlich gemacht werden, wo diese Ansätze Defizite aufweisen oder nicht anwendbar sind, z. B. weil sie von anderen Umgebungen oder Voraussetzungen ausgehen.

Da heutzutage beinahe jedes Gerät ein Vibrationsmotor verbaut hat, sei es das Handy, die Smartwatches oder Fitnessarmbänder (uvm.), werde ich im folgenden noch auf einige aktuelle Technologien und deren Umsetzung der personalisierten Vibrationsmuster zu sprechen kommen.

2.3.1 Taptic Engine

Die Taptic Engine ist eine von der Firma Apple selbst entwickeltes Vibrationsmotor, dass heutzutage in nahezu allen Apple Produkten verbaut ist. Das erste Gerät was die Taptic Engine bekommen hat, war die Apple Watch. Der Name **Taptic** bildet sich aus dem Wörtern "Taktild und "Haptisch". Trotz des neu erfinden einer mechanischen Rückmeldung bietet Apple keine Personalisierung, wie lange eine Rückmeldung, für die Apple Watch, erfolgen soll. Die Einstellungsmöglichkeiten an der Apple Watch ist lediglich die Stärke der Vibration. Diese ist in 3 Stärkestufen unterteilt. Meiner Ansicht nach, kann man daher nicht wirklich von einer personalisierten Vibration sprechen.

2.3.2 Personalisierte Vibration

Der Hersteller Apple hat auch bei dem iPhone eine Möglichkeit geboten eigene Vibrationsmuster zu erstellen, jedoch mit Einschränkungen. Wenn man in die jeweilige Einstellung der iPhones gelangt, erscheint das folgende Bild. Beim drücken auf das Display wird an der Stelle eine Vibration erzeugt. Man hat 10 Sekunden um ein eigenes Muster zu erzeugen, indem man wiederholt auf den Bildschirm drückt. An der Stelle, an der man den Bildschirm berührt hat, erscheint visuell um der Position ein Kreis. Die erzeugten Vibrationen werden in einer Leiste visuell angezeigt. Man kann sich beliebig viele Vibrationsmuster speichern, die bis zu 10 Sekunden lang sind. [FSDS16]

Die Einschränkung die man hier nennen muss ist, dass man die Vibrationsmuster nur für Systeminterne Funktionen benutzen kann. Dies bedeutet, dass man die Funktionen für Klingeltöne, Nachrichtentöne, Erinnerungshinweise, Kalenderhinweise (o.ä.) hinzufügen kann. Für eine andere Anwendung, die nicht im Betriebssystem integriert ist, ist das nicht möglich. Somit können Benachrichtigungen von anderen Entwicklern keine eigene Vibrationsmuster erhalten. Daraus folgt, wenn das iPhone in der Hosentasche ist und ich eine Benachrichtigung von einer Application erhalte, die nicht im System integriert gewesen ist, kann man anhand der Vibrationen des iPhones nicht unterscheiden welche Application dies gewesen ist.



Abbildung 2.1: Settings on the apple watch

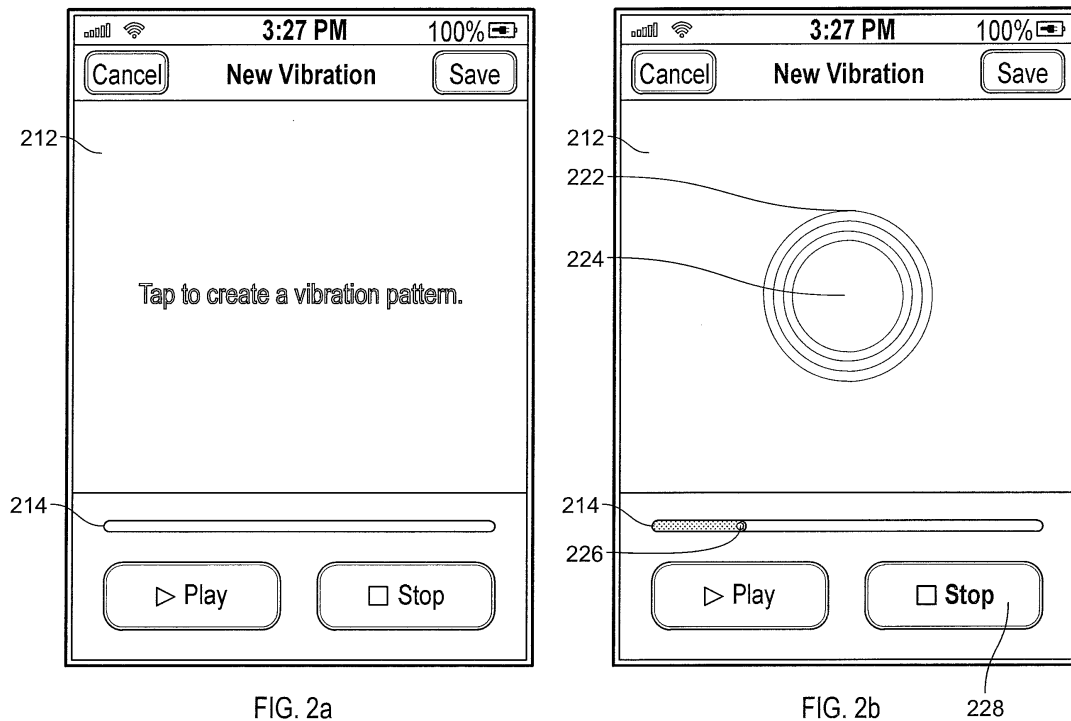


FIG. 2a

FIG. 2b

228

Abbildung 2.2: custom vibrations on the iPhone

2.3.3 Personalisierte Smartwatch

Das Gerät, dass es nach meiner Recherche am besten gelöst hat, ist eine Smartwatch von einem kleinen StartUp namens Martian. Das Startup hat eine Uhr hergestellt, mit der man mittels einer App auf dem Smartphone die Vibrationsmuster selbst anpassen kann. Die App unterstützt eine große Anzahl an Applications, von anderen Herstellern, die Benachrichtigungen senden. Ein Vibrationsmuster für die Uhr kann man aus mit zu 4 Signalen auf der Uhr darstellen lassen. Die Signale sind als Lang, Kurz und Pause festgelegt. Somit kann man mittels der Vibration der Uhr herausfinden, welche App gerade eine Benachrichtigung auf mein Handy gesendet hat. Die Länge und Stärke eines Signals ist schon im Vorfeld festgelegt.

2.3.4 Fazit

Bei sehr vielen Herstellern ist es aktuell noch gar nicht möglich eigene Vibrationsmuster zu erstellen. Bei Android Geräten ist es aktuell so, dass man nur aus einer Menge von wenigen Vordefinierten Vibrationsmustern sich nur eines auswählen kann. Einige Entwickler haben dieses Problem erkannt und eine eigene Applikationen entwickelt und im Store veröffentlicht..

(BILD)

2.3.5 Design of a Wearable Tactile Displays

In dem Paper von XXX [] handelt es um Taktile Displays; was man bei der Erschaffung von Taktilem Display beachten soll und was für Verwendungszwecke es noch gibt. Das Paper ist schon mehr als 15 Jahre alt und weist dennoch auf Informationen hin, die heutzutage noch Relevant sind. Die Inhalte des Papers werden im folgenden beschrieben. Taktile Displays sind Geräte, die dazu benutzt werden um Informationen durch Körperkontakt des



Abbildung 2.3: possible settings on the martian watch

Menschen über Haptisches Feedback zu übermitteln. Diese bilden keinen Konflikt mit den audio-/visuellen Einflüssen. Die Taktilen Displays sind eine Unterstützung der Darstellung der Informationen. Beispielsweise kann man Blinden oder Tauben Informationen mittels Taktilen Displays vermitteln, die Sie nicht wahrnehmen können. Dabei werden haptische / sensorische assistive Geräte benutzt um die Informationen in der Realen Umgebung in taktile Simulationen umzuwandeln. Ein wichtiger Aspekt im Paper ist es gewesen, wie man ein solches Taktilen Display entwirft und welche Aspekte man beachten soll. Beim Design eines Taktilen Devices sollten leise und leicht, klein sein; wenig Strom verbrauchen; die Taktoren sollten durch Kleidung spürbar sein und am besten so eng wie möglich am Körper liegen. Handys schon damals Vibrationsmotoren gehabt um einem Nutzer darauf Aufmerksam machen wollte, dass man eine Nachricht erhalten hatte. Die Vibration des Handys war eine Metapher dazu, dass eine andere Person einem an die Schulter rütteln würde. [] Um es nicht nur Theoretisch zu erklären hat man ein Taktilen Display als Weste entworfen, bei dem man die Vibrationsmotoren aus Nokia Handys verwendet hat. Mittels der Weste sollte eine Person vom Punkt A zu Punkt B navigiert werden. Die übermittelten Informationen zur Navigation sind vorwärts, zurück, rechts, links, beschleunigen und verlangsamen gewesen.

3. Analyse

In diesem Kapitel sollten zunächst das zu lösende Problem sowie die Anforderungen und die Randbedingungen einer Lösung beschrieben werden (also nochmal eine präzisierte Aufgabenstellung).

Dann folgt Üblicherweise ein Überblick über bereits existierende Lösungen bzw. Ansätze, die meistens andere Voraussetzungen bzw. Randbedingungen annehmen.

3.1 Anforderungen

Anforderungen und Randbedingungen ...

Die Aufgabenstellung bestand darin, herauszufinden, ob ein personalisiertes Vibrationsmuster besser als ein vorgegebenes Vibrationsmuster erkannt wird. Man sollte in dieser Bachelorarbeit ein System bauen, dass mit dem Wearable kommuniziert und Daten übertragen kann, so dass das Wearable Vibrationen abspielt. Darüber hinaus soll eine Studie erstellt werden, in der man herausfindet, ob es einen signifikanten Unterschied zwischen generischen und genetischen Vibrationen gibt. Dabei sind die generischen Vibrationen fest vorgegeben, wobei die genetischen Vibrationen für den Probanden so angepasst werden sollen, dass diese genau für ihn angepasst werden. Dabei habe man ein Evolutionären Algorithmus verwendet um die genetische Vibration für einen Probanden zu bestimmen.

Um dies zu lösen musste man sich im Vorfeld ein paar Gedanken über die Repräsentation eines Signals, die Dekodierung und über die Übertragung machen. Im Folgenden werden diese Entscheidungsfindungen beschrieben.

Wearable

Damit man Vibrationen überhaupt darstellen kann, hätte man sich ein eigenes Taktils Device entwerfen müssen. In meinem Fall war dies nicht Nötig, da das Wearable vom TECO vorgegeben war. Die Komponenten des Wearables sind zwei Taktoren (TI TLC5971), einen Mikroprozessor (nRF51822 / BLE Nano) und eine Batterie.

Die Kommunikation des Armbands konnte man über eine Serielle Schnittstelle, sowie über Bluetooth Low Energie (LE) realisieren. Bei der seriellen Schnittstelle ist ein USB-Kabel nötig gewesen um den PC mit dem Wearable anzusprechen, dabei wäre die Mobilität verloren gegangen. Daher habe man die verfügbare BLE Schnittstelle benutzt um zwischen dem PC und dem Wearable zu kommunizieren.

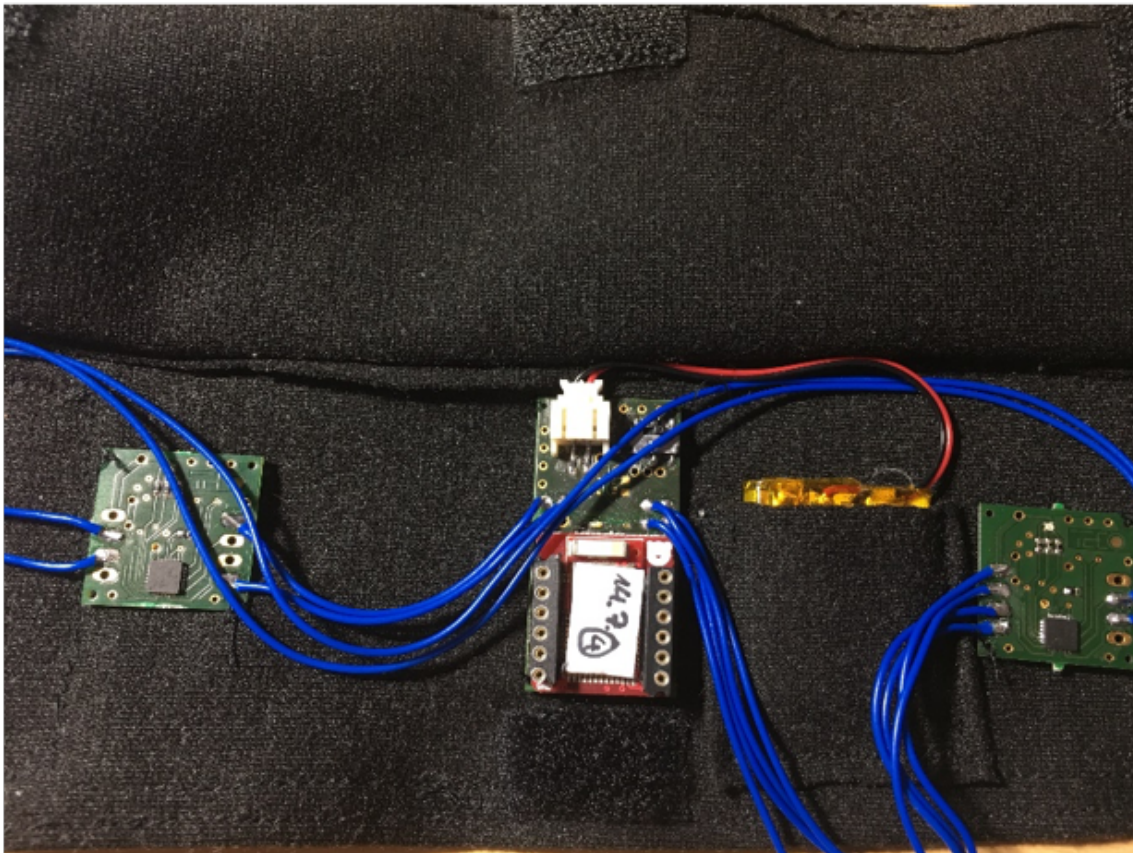


Abbildung 3.1: Armband angezogen



Abbildung 3.2: Armband geöffnet. Komponenten 2 Traktoren, Batterie, und Mikroprozessor

Bluetooth LE überträgt Nachrichten nur in einer maximalen Größe von 20 Bytes. Diese werden jeweils als zwei Bytes Blöcke übertragen.

Anhand der Begrenzung der Datenübertragung hat man sich eine geeignete Dekodierung des Signals überlegen müssen.

Die Programmiersprache um die Kommunikation zwischen dem PC und dem Wearable aufzubauen, konnte frei gewählt werden. Man habe sich hierbei zuvor verschiedenste Programmiersprachen angeguckt. Java hat man nach fehlenden Informationen zum Kommunikationsaufbau ausgeschlossen. Man hat sich letztendlich für C# entschieden, da man eine funktionierende Kommunikation aufgebaut hat und man gleichzeitig die Grafische Oberfläche nutzen kann um Studie darin durchzuführen.

Dabei musste herausfinden was für Einstellungsmöglichkeiten man am Wearable hat. Diese waren die Länge und die Stärke eines Signals.

Signale

Ein Signal repräsentiert eine Vibration, die auf dem Wearable abgespielt wird. Es hat sich die Frage gestellt, wie personalisiert man denn jetzt Signale. Dabei hat man das Signal in drei verschiedene Längen definiert, Kurz, Mittel und Lang.

Dabei hat man als Vorgabe gehabt, dass man einen Evolutionären Algorithmus benutzt. Diesen EA hat man verwendet, um für die Signaltypen ein personalisierten Wert bestimmen.

Dabei hat man sich ein Vorgang überlegen müssen, um für den EA eine Startpopulation zu erzeugen, die Individuen der Population zu bewerten, mittels einer Fitnessfunktion den Fitnesswert für jedes Individuum berechnen, eine geeignete Selektion, Reproduktion und Mutation bestimmen, sowie dass die Terminierungsbedingung.

Eine geeignete Signalrepräsentation musste definiert werden um Signale vom PC zum Wearable über BLE zu senden und abspielen zu lassen. Das bedeutet sowohl am PC als auch am Wearable selbst musste dies definiert werden.

Studie

Es sollte eine Grafische Oberfläche erstellt werden, damit der Benutzer Studie komplett über das Programm durchführen kann indem er die Instruktionen des Programms folgt und er auch selbst entscheiden kann, wann er ein Signal abgespielt haben will.

Es hat sich durch das Definieren der Signale und dem EA ergeben, dass die Studie in drei Schritten unterteilt wird. Der erste Schritt ist es, die Grenzen für eine Startpopulation für einen EA zu bestimmen. Der zweite Schritt ist es, einen genetischen Wert für die eigentliche Studie zu bestimmen, dies passiert durch ausführen mehrerer Generationen des EA. Der letzte Schritt ist es, generische und genetische Signale vom Benutzer bewerten zu lassen.

Alle Daten sollten überprüft werden und es sollte herausgefunden werden ob es am signifikante Unterschiede gibt.

3.2 Existierende Lösungsansätze

Hier kommt eine ausführliche Diskussion von „Related Work“.

Abbildungen sollten möglichst als EPS (Encapsulated Postscript) bzw. PDF eingebunden werden. Zur Erzeugung sauberer EPS-Dateien empfiehlt sich das Tool **ps2eps** zur Nachbearbeitung von Postscript-Dateien. Mit **epstopdf** kann dann eine PDF-Datei zum Einbinden erzeugt werden.

Abbildung 3.3: Testabbildung

3.3 Weiterer Abschnitt

Bla faasel... hat auch schon [?] gesagt und [?, ?, ?] sollte man mal gelesen haben. Abbildung 3.3 auf S. 14 sollte man sich mal anschauen.

Blindtext Blindtext Blindtext Blindtext Blindtext Blindtext Blindtext Blindtext Blindtext
Blindtext Blindtext Blindtext Blindtext Blindtext

Blindtext Blindtext Blindtext Blindtext Blindtext Blindtext Blindtext

3.4 Zusammenfassung

Am Ende sollten ggf. die wichtigsten Ergebnisse nochmal in *einem* kurzen Absatz zusammengefasst werden.

4. Entwurf

In diesem Kapitel erfolgt die ausführliche Beschreibung des eigenen Lösungsansatzes. Dabei sollten Lösungsalternativen diskutiert und Entwurfsentscheidungen dargelegt werden.

4.1 Programm Ablauf

4.2 Ausführung des Programms

Zu Beginn hat man sich über das gesamte Problem einen Überblick schaffen müssen. Dabei hat man sich alle einzelnen Bestandteile im Detail angeguckt, um herauszufinden welche Komponenten man wie zusammensetzt. Es haben sich drei / vier große Komponenten ergeben, die man erledigen musste.

Zuerst habe man sich mit dem Wearable auseinandergesetzt, um zu bestimmen, was man alles mit dem Wearable umsetzen kann. Man fand heraus, dass man zwei Parameter hatte die man einstellen konnte. Diese waren die Zeit, wie lang das Wearable vibriert und die Stärke, wie stark das Armband vibriert. Um ein Signal zu übertragen hatte man eine Beschränkung von 20 Bytes, die man maximal in einer Nachricht mittels Bluetooth übertragen kann.

Als nächste Komponente des Problems habe man sich eine geeignete Repräsentation eines Signals überlegen müssen. Um nicht außer Bedacht zu lassen ist diese die wichtigste Komponente des Problems, denn es hängen alle anderen Komponenten davon ab, wie diese Daten repräsentiert werden.

Daher stellte sich die Frage, wie sieht ein solches Signal aus?

4.2.1 Signal

Anhand der zwei Parameter, die Länge und die Stärke des Wearables, hat man diese als Attribute eines Signals definiert.

Technisch gesehen hätte man für die Länge eines Signals von 0 ms (0x0000) bis 65535 ms (0xFFFF) nutzen können. Man hat sich jedoch anhand vorheriger Studien [?] orientiert und ein Minimum von 100 ms und ein Maximum von 1000 ms als Grenzen für die Länge übernommen.

Bei der Stärke eines Signals hat man sich auch hier am Wearable orientiert. Theoretisch gesehen hätte man die gleichen Grenzen von 0 bis 65535 zur Repräsentation der Stärke

vorhanden, aber technisch hat man erst ab einer Stärke von 32767 (0x7FFF) ein spürbare Vibration gehabt. Vor dem genannten Wert ist zu wenig Strom übertragen worden, sodass kein Traktor funktionierte. Daher hat man sich zwischen den Grenzen von (0x7FFF) und (0xFFFF) für die Stärke festgelegt.

Anhand der Hypothese der Bachelorarbeit wolle man wissen, wie gut sich personalisierte Vibrationen im Vergleich zu generischen Vibrationen verbessern. Dabei musste man sich überlegen, wie man personalisierte Vibrationen mittels dem EA für einen Probanden bestimmen will. Außerdem sollte man herausfinden was für Signale noch gut voneinander unterscheidbar sind, die die Grenzen von 100ms bis 1000ms besitzen. Man habe sich auf drei Typen von Signalen festgelegt: **Kurz**, **Mittel** und **Lang**. Dadurch wolle man wissen, ob ein Signal als Kurz, Mittel oder Lang empfunden wurde.

Die jeweiligen Typen definieren innerhalb der Grenzen zwischen 100 ms und 1000 ms ein Intervall, die sich nicht überschneiden. Abgesehen davon ist **kurz**, das zahlenwertigste kleinste Intervall, gefolgt von **mittel**, **lang** ist das zahlenwertigste größte Intervall. Allerdings wurde es darauf geachtet, dass die Grenzen nicht aufeinander liegen, sondern einen Abstand zwischen den Grenzen existiert. Als Beispiel habe man für Kurz die Intervallgrenzen 100ms und 300ms, für Mittel habe man dann die Intervallgrenzen von 400ms bis 600ms und für Lang habe man die Intervallgrenzen von 700ms bis 1000ms.

Für die Stärke der Vibration war man an den Darstellbarkeit des Wearable gebunden. Zwischen die Grenzen von (0x7FFF) und (0xFFFF) hat man sich fünf voneinander unterscheidbare Stufen definiert. Für den späteren Verlauf habe man sich Zustände definiert.

Signalstärke	Namen	Zustandsnamen
0x7FFF	Very Weak	q_0
0x9FFF	Weak	q_1
0xBFFF	OK	q_2
0xDFFF	Strong	q_3
0xFFFF	Very Strong	q_4

4.3 Evolutionärer Algorithmus

4.4 Evolutionärer Algorithmus

4.4.1 Eingabe für den Algorithmus

Grenzen Initialisieren

Als Eingabe für den Algorithmus benötigt man eine Start Population / Anfangspopulation von Individuen. Die Individuen sind in diesem Fall Signale. Nicht jede Person empfindet ein vorgegebenes Kurzes Signal gleich wie eine andere Person, somit musste man zuvor den Benutzer befragen, was er als Kurz, Mittel und Lang empfindet. Zuerst habe man alle Signale abgespielt, damit der Proband wusste, was Ihn erwartet. Nachdem alle Signale abgespielt wurden, wurde jedes Signal einzeln abgespielt und der Proband hatte die Aufgabe das abgespielte Signal der Kategorie Kurz, Mittel oder Lang zuzuordnen. Jedem Probanden wurde insgesamt 10 Signale mit der gleichen Vibrationsstärke abgespielt, die Signalelänge ist dabei gleich verteilt. Dabei wurden alle 10 Signale in einer zufälligen Reihenfolge abgespielt.

Nach der Eingabe des Probanden erhält man beispielsweise folgende Bewertung. Diese Eingabe ist ideal, da alle Signaltypen direkt hintereinander vorliegen, so würde man hier ein die Grenzen sofort aus der Tabelle entnehmen können. Diese belaufen sich für Kurz zwischen 100 und 300 ms, für Mittel zwischen 400ms und 600ms und für Lang zwischen 700 und 1000ms.

Tabelle 4.1: My caption

Signallänge (in ms)	100	200	300	400	500	600	700	800	900	1000
Erkannten Signaltyp	Kurz	Kurz	Kurz	Mittel	Mittel	Mittel	Lang	Lang	Lang	Lang

Tabelle 4.2: My caption

Signallänge (in ms)	100	200	300	400	500	600	700	800	900	1000
Erkannten Signaltyp	Kurz	Kurz	Mittel	Mittel	Lang	Mittel	Lang	Lang	Lang	Lang

Falls die Eingabe jedoch nicht so Ideal sein sollte, wie in dem Beispiel gerade eben, mussten ein paar Vorkehrungen getroffen werden. Um einige Sonderfälle auszuschließen, hat man überprüft, ob das Signal mit der Länge von 100ms ein Kurzes Signal ist und das Signal mit der Länge von 1000ms ein Langes Signal ist, sowie man annimmt, dass mindestens jeder Signaltyp mindesten zwei mal ausgewählt wurde. Sollte dies nicht der Fall sein, so würde man den Benutzer dazu bitten, die zehn Signale erneut zu bewerten.

Man beginnt damit die neuen Intervallgrenzen zu bestimmen. Diese wurden anhand des Beispiels exemplarisch in der Tabelle bestimmt, dabei ist jede Spalte eine Iteration.

Bei einer kleinen Abweichung von zwei nebeneinanderliegen von zwei Werten ist dies noch akzeptabel, bei größeren Abweichungen, hat man Benutzer noch einmal darum gebeten erneut zu bewerten. Im Verlauf der Studie musste man nur bei einer Minderheit von Probanden ein weiteres mal darum bitten, die Signale neu zu bewerten. Denn es ist oft schon so gewesen, dass die Probanden es wie im Idealfall zugeordnet hatten.

Population

Nachdem man jetzt die Grenzen für einen Probanden bestimmt hat, kann man endlich die Startpopulation des Algorithmus erzeugen. Bei der Startpopulation hat man für jeden Signaltypen zehn Signale. Die zehn Signale beinhalten zwei Signale, die das Minimum und Maximum des jeweiligen Signaltypen repräsentieren. Die restlichen acht Signale erhalten eine zufällige Länge innerhalb des Intervalls. Die Stärke eines Signal wird zufällig für jedes Signal zufällig zugewiesen. Man besitzt somit für die Startpopulation dreißig Signale, die jeweils in zehn Kurze, Mittlere und Lange Signale unterteilt sind.

Bewertung des Algorithmus

Jedes Individuum der Startpopulation muss für die Fortsetzung des nächsten Schritts vom Algorithmus bewertet werden. Um diese Bewertung zu erhalten hätte man aus einigen Alternativen wählen können wobei ich hier ein weiteres erläutern werde und wieso man dieses nicht gewählt hat.

Tabelle 4.3: My caption

Iterationen	1.	2.	3.	4.	5.	6.	7.	8.	10.
$Kurz_{Min}$	100	100	100	100	100	100	100	100	100
$Kurz_{Max}$	100	200	200	200	200	200	200	200	200
$Mittel_{Min}$	100	200	300	300	300	300	300	300	300
$Mittel_{Max}$	100	200	300	400	400	600	600	600	600
$Lang_{Min}$	100	200	300	400	500	600	700	700	700
$Lang_{Max}$	100	200	300	400	500	600	700	800	1000

Tabelle 4.4: My caption

Antwort	sehr schlecht	schlecht	ok	gut	sehr gut
Wertigkeit	-2	-1	0	+1	+2

Tabelle 4.5: My caption

Antwort	sehr schlecht	schlecht	ok	gut	sehr gut
Wertigkeit	1	2	3	4	5

Zuerst könnte man hergehen und jedes Signal aus der Population dem Probanden abspielen und fragen was für ein Signaltyp er erkannt hat, wie man es bei den ersten zehn Signalen gemacht hat um die Grenzen der Signaltypen zu bestimmen.

Der erste Ansatz wäre man würde genau dies tun, dabei würde man die Anzahl der richtigen Zuweisungen und Abweichungen zählen um mit einer geeigneten Fitnessfunktion den Fitnesswert zu bestimmen. Ferner würde das bedeuten, dass für eine Population ein Individuum mehrmals abgespielt werden müsse um eine Bewertung des einzelnen Individuums zu erhalten. Bei einer Annahme von fünf Bewertungen pro Individuum, um nur eine Population damit bewerten zu können, müsste der Proband eine Anzahl von 150 Signalen bewerten, damit der Algorithmus eine Generation bestimmen kann. Dies würde bedeuten, dass wenn man vier Generationen bestimmen wollen würde, man auf eine Anzahl von 600 Bewertungen alleine für die Bestimmung des personalisierten Wertes mithilfe dem EA.

Aufgrund der Tatsache, dass Probanden nicht so lange an einer freiwilligen Studie teilnehmen wollen, habe man sich für einen anderen Ansatz entschieden. Man muss akzeptieren, dass man über mehrere Generationen hinweg die komplette Population bewerten muss. Das heißt, dass man pro Generation 30 Signale abspielen muss. Um eine Generation zu bestimmen habe man sich Gedanken darüber gemacht, wie man so ein Signal bewerten soll. Dabei ist man zu dem Entschluss gekommen, dass man man den Benutzer zuerst fragt, was für ein Signaltyp er erkannt habe. Weiterhin habe man ihn zwei Fragen wie in einem Fragebogen (Likert-Skala) gefragt, wie gut er das Signal erkannt wurde und wie man die Stärke des Signals empfunden hat.

Wie oben schon erwähnt, für die Anfangspopulation hat jedes Individuum eine zufällige Stärke zugewiesen. Dabei entspricht die Stärke einem Zustand aus dem Zustandsdiagramm Y. Anhand der Bewertung verändert sich der aktuelle Zustand der Stärke eines aktuellen Individuums um bis zu zwei Zustände. Dabei ist die Wertigkeit der Antwort der Frage in der Tabelle XX beschrieben. Dabei bedeutet +2, dass der aktuelle Zustand zwei mal +1 im Zustandsdiagramm XW ausführt.

Die Antwort was für ein Signal erkannt wurde zeigt, was für ein Signal der Proband erkannt hat. Dies ist für die Bewertung der Studie hilfreich. Die letzte Frage die der Benutzer zu beantworten hat, wie er das Signal erkannt hat, hat die Wertigkeit wie in Tabelle XA definiert. Diese Antwort auf die Frage, ist der entscheidende Wert, der den Fitnesswert repräsentiert.

Erzeugung der nächsten Generation

Im Anschluss nach der Bewertung erfolgt die Bestimmung der nächsten Generation. Dies erfolgt mit der Selektion, Rekombination und Mutation der Gene.

Selektion

Für die Selektion wird für jeden Signaltyp ein neuer Pool erzeugt. Im Anschluss daran werden die drei Maxima für jeden Signaltypen von allen Fitnesswerten bestimmt. Anhand

der Formel ($\text{Fitnesswert eines Individuums} / \text{Maximum des zugehörigen Signaltyps}$) * 100 lässt sich die Häufigkeit bestimmen, wie oft ein Individuum in den Pool hinzugefügt wird. Als Beispiel hat man ein Maximum von 5 und einen Fitnesswert von 2, daraus ergibt sich 40, das heißt, es wird 40 mal das jeweilige Individuum in den Pool hinzugefügt. Dabei wird jedes Individuum in das Pool hinzugefügt, welchem Signaltyp es angehört.

Nach der Erzeugung der Pools wird eine Rekombination durch zwei zufällig ausgewählte Individuen ausgeführt.

Rekombination

In der Rekombination wurden die zwei zufällig ausgewählten Individuen als Eltern definiert. Die gefundenen Eltern erzeugen ein neues Individuum, dass in die Population der nächsten Generation hinzugefügt wird.

Dieser Vorgang wurde wie folgt für das Problem angepasst. Man hat die zwei Gene, die Stärke und die Länge eines Signals.

Zuerst beginnen wir mit der Länge des Signals. Es wird die Länge des ersten Elternteils und die Länge des zweiten Elternteils genommen und ein Mittelwert von beiden Werten gebildet. Der Mittelwert ist die Länge des neuen Individuums. Für die Stärke des neuen Individuums wird ebenfalls der Mittelwert von beiden Elternteilen gebildet und abgerundet.

Aus jedem der drei erzeugten Pools werden zehn neue Individuen für die nächste Generation erzeugt, so dass man wieder auf eine Gesamtzahl von 30 Individuen für eine Population kommt.

Mutation

Es besteht eine Chance von 5% dass sich ein Individuum mutieren kann. Das bedeutet in diesem Fall, dass sich lediglich die Länge zufällig verändern kann.

Abbruchkriterium

Man wiederholt für jede Generation wie im (BILD) dargestellten Zyklus, bis man eine ausreichende Lösung erhält.

Das Abbruchkriterium ist erreicht, wenn die vierte Generation erzeugt worden ist. Um hier die Stellung zu der Entscheidung von vier Generationen zu nehmen, ist es so, dass man versucht habe die Anzahl der Vibrationen für einen Probanden niedrig zu belassen. Im Vergleich zum ersten Ansatz, in der man 600 Vibrationen für vier Generationen hat, habe man für die aktuell benutzte Lösung bei 30 Individuen pro Population, für 4 Generationen, 120 Vibrationen.

Nach dem man die letzte Population vom Algorithmus erhält, bestimmt man für jeden Signaltypen das Minimum und Maximum von der Stärke und der Länge von allen Signalen. Anhand derer man den Mittelwert bestimmt und schließlich den personalisierten Wert für den Probanden erhält.

4.4.2 Muster

Ein Muster ist eine Folge von mehreren Signalen die hintereinander abgespielt werden.

Damit man schließlich die Hypothese der Bachelorarbeit beantworten kann, muss man zuvor die generische Vibrationen mit genetische Vibrationen vergleichen. Man habe im Vorfeld 15 Muster eine Signallänge von drei, vier und fünf, dass macht in der Summe

45 Muster. Man habe diese 45 Muster zwei mal erzeugt, jedoch einmal mit den Kurz, Mittel und Lang Werten und die dazugehörige Stärke, die man nach dem Algorithmus bestimmt hat und einmal fest Vorgegebene Werte von Kurz, Mittel und Lang sowie die Stärke *Strong*. Bei den Vordefinierten Werten hat man sich wieder anhand dem Paper [?] für einen Kurz Wert von 200ms, einen Mittel Wert von 400ms und einem Lang Wert von 800ms orientiert.

Dabei habe man den Probanden in der Studie dabei gefragt, was für Signale er in welcher Reihenfolge erkannt habe. Diese Daten wurden in der Evaluierung ausgewertet.

4.5 Zusammenfassung

Am Ende sollten ggf. die wichtigsten Ergebnisse nochmal in *einem* kurzen Absatz zusammengefasst werden.

5. Implementierung

Bla fasel...

5.1 Signal

Bla fasel...

Das Signal ist wie im Entwurf schon beschrieben, einer der wichtigsten Bestandteile meines Programms. Jedes Signal beinhaltet die Attribute Länge, Stärke, der Signaltyp sowie die Grenzen des Signaltyps.

Je nach dem in welcher Phase des Programms man sich gerade befindet, werden noch weitere Attribute gespeichert. Das Programm besteht aus vier Phasen. 1. Aufnahme der Personalien 2. Bestimmung der Grenzen durch Bewertung durch den Nutzer 3. Trainingsphase:

Die wichtigsten Parameter die hier gespeichert werden hängen davon ab, in welchem Schritt des Programms man sich gerade

Das Signal ist ein wichtiger Bestandteil meines Programm. Ein Signal beinhaltet die Signallänge, die in Millisekunden gespeichert wird, und eine Signalstärke, die in 5 Stärkestufen eingeteilt ist.

Ich habe meinen Evolutionären Algorithmus so angepasst, dass bei mir ein Individuum ein Signal ist. Ich habe dem Benutzer das Signal mit dem Wearable abspielen lassen und im Anschluss Fragen beantworten lassen. Er sollte bewerten wie gut er das Signal erkannt hat. Die Bewertung vom Benutzer war entscheidend um nach der kompletten Bewertung der Population

5.2 Muster

Bla fasel...

5.3 Evolutionärer Algorithmus

Bla fasel...

6. Evolutionärer Algorithmus

6.1 DNA

6.2 Population

6.3 Generationen

6.4 Mutation

6.5 Studiendesign

Bla fasel. . .

6.5.1 Studiendesign

Im folgenden Benutzerflussdiagramm hat man sich orientiert um die Studie zu entwerfen. (Sample of convenience) Über Mailinglisten und Bekanntenkreis haben sich 32 Probanden bereiterklärt an der Studie teilzunehmen. Dabei waren 72 Prozent Männer und 28 Prozent Frauen. Das Alter der Probanden war zwischen 12 bis 54 Jahren vertreten und das Durchschnittsalter war 22 Jahre. Die Studie hat zwischen 30 Minuten und einer Stunde gedauert. Die Studien wurden an drei verschiedenen Orten durchgeführt, im TECO in Karlsruhe, in einem Seminarraum an der Hochschule Darmstadt und in einem Arbeitszimmer in Meschede.

Für jeden teilgenommenen Probanden ist der gleiche Ablauf durchgeführt. Vor der Studie wurde ein Termin mit dem Probanden vereinbart. Nachdem der Proband zur abgemachten Zeit am vorgegebenen Ort angekommen ist, wurde Ihm erklärt wofür die Studie ist, was man mit der Studie herausfinden will und welche Erwartungen man an den Probanden hat. Nachdem der Proband alles verstanden hat und die Einverständniserklärung verstanden und unterschrieben hat, wurde Ihm das Armband angezogen.

Für die Studie hat man ein Programm mit einer Grafischen Oberfläche (GUI) entworfen, mit der es möglich war die ganze Studie durchzuführen. Dabei wurden dem Probanden ein paar Personalien abgefragt, wie das Alter, das Geschlecht, ob sich die Person als Musikalisch empfindet, ob man Computerspiele spielen würde, ob man schon einmal eine Smartwatch benutzt habe und ob die Person schon mal ein Tactiles Gerät benutzt habe.

Falls vom Probanden Fragen während der Studie Fragen aufgekommen sind wurden diese sofort beantwortet.

Nach der Aufnahme der Personalien, wurde dem Benutzer erklärt, was Ihn als nächstes erwartet und von Ihm verlangt wird. Man hat dem Nutzer im ersten Schritt 10 Signale abgespielt, um Ihn ein Gefühl für Signale zu geben. Im Anschluss wurde dem Probanden jedes Signal erneut einmalig abgespielt. Dabei sollte er das Signal zu drei jeweiligen Kategorien zuordnen. Diese Kategorien waren ob es ein Kurzes, Mittleres oder Langes Signal für Ihn gewesen ist. Dieser Schritt war dafür notwendig um für den Benutzer die Grenzen für die jeweilige Kategorien Kurz, Mittel und Lang zu bestimmen. Diese Grenzen sind für die Initialisierung des Algorithmus notwendig gewesen.

Als die 10 Signale bewertet wurden, wurde der Benutzer darüber aufgeklärt, was Ihm als nächsten Schritt erwartet. Es wurde Ihm ein anhand seiner Eingaben ein zufälliges Signal abgespielt, dass er bewerten sollte (BILD). Anhand der drei Fragen wurde das Signal bewertet. Um eine Iteration komplett zu bewerten wurde dieser Vorgang 30 mal wiederholt. Im Anschluss wurde gefragt wie der Benutzer sich derzeit fühlt (BILD). Anhand einer komplett bewerteten Iteration wurde dem Benutzer neue Werte berechnet. Es wurden insgesamt vier Iterationen durchgeführt um einen möglichst genauen Wert für den Benutzer zu bestimmen.

Im letzten Schritt wurde dem Benutzer aufgeklärt, dass ab dem Zeitpunkt nur noch Folgen von Signalen, die man ab jetzt Muster nennt, abgespielt werden. Die Probanden sollten angeben in welcher Reihenfolge was für Signaltypen abgespielt wurden. Es wurde für alle Probanden im Vorfeld alle Muster definiert, damit jeder die gleichen Muster abgespielt bekommt. Es gab zwei Arten von Muster, die generischen Muster und die genetischen Muster. Der einzige Unterschied zwischen den beiden Arten waren die Werte, die die Signale in einem Muster zugewiesen wurden. Das bedeutet es wurden zwei mal das selbe Muster abgespielt mit lediglich anderen Werten. Die Genetischen Muster hatten die Werte, die nach dem Algorithmus erzeugt wurden übernommen, wobei die generischen Muster einen vordefinierten Wert zugewiesen bekommen hat. Der generische Wert ist für jeden Probanden gleich gewesen. Dabei gab es Muster mit drei, vier und fünf Signalen. Nacheinander wurde dem Nutzer zuerst alle Muster mit drei Signalen. Dabei wurde das genetische Muster abwechselnd zum generischen Muster abgespielt.

Nachdem alle Muster von dem Probanden bewertet wurden, haben Sie Ihre e-Mail noch angegeben um an einer automatischen Verlosung von zwei Gutscheinen teilzunehmen. Bei Interesse wurde Ihnen Ihre Werte gezeigt und erklärt, was genau im Hintergrund passiert worden ist. Während der ganzen Studie standen dem Probanden ausreichend Süßigkeiten zur Verfügung, bei denen Sie sich frei bedienen konnten.

7. Evaluierung

Hier kommt der Nachweis, dass das in Kapitel 4 entworfene Konzept auch funktioniert. Leistungsmessungen einer Implementierung werden auch immer gerne gesehen.

7.1 Initialisierung der Grenzen

7.2 Abschnitt 2

7.3 Zusammenfassung

Am Ende sollten ggf. die wichtigsten Ergebnisse nochmal in *einem* kurzen Absatz zusammengefasst werden.

8. Zusammenfassung und Ausblick

Bla fasel...

(Keine Untergliederung mehr!)

9. Zusammenfassung und Ausblick

Bla fasel...

(Keine Untergliederung mehr!)

Literaturverzeichnis

- [app18] *Apple Watch Settings*, 2018 (accessed May 1, 2018). <https://support.apple.com/de-de/HT204793>.
- [Fli] Patrick Flick: *Evolutionäre Algorithmen*.
- [FSDS16] Christopher B Fleizach, Eric Taylor Seymour und Joel M Lopes Da Silva: *Custom vibration patterns*, jul # -5" 2016. US Patent 9,383,820.
- [GOS01] Francine Gemperle, Nathan Ota und Dan Siewiorek: *Design of a wearable tactile display*. In: *Wearable Computers, 2001. Proceedings. Fifth International Symposium on*, Seiten 5–12. IEEE, 2001.
- [Sel03] Bianca Selzam: *Genetische Algorithmen*. Dortmund: TU Dortmund, 2003.
- [Shi12] Daniel Shiffman: *The Nature of Code: Simulating Natural Systems with Processing*. Daniel Shiffman, 2012.

Erklärung

Ich versichere, dass ich die Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht und die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet habe. Die Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt und von dieser als Teil einer Prüfungsleistung angenommen.

Karlsruhe, den 17. Mai 2018

(thomas)