



FAKULTÄT
Informatik, Mathematik und Naturwissenschaften

VISUALISIERUNG KONTINUIERLICHER, MULTIMODALER SCHMERZ SCORES AM BEISPIEL AKUSTISCHER SIGNALE

Abschlussarbeit zur Erlangung des akademischen Grades
MASTER OF SCIENCE

Studiengang MEDIENINFORMATIK

Autor:
Franz ANDERS

Erstbetreuer HTWK:
Prof. Dr. Mario HLAWITSCHKA

Zweitbetreuer LaBP:
Mirco FUCHS

19. März 2017

Eidesstattliche Versicherung

Ich erkläre hiermit, dass ich diese Masterarbeit selbstständig ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel verfasst habe. Alle den benutzten Quellen wörtlich oder sinngemäß entnommenen Stellen sind als solche einzeln kenntlich gemacht. Diese Arbeit ist bislang keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht worden. Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

Leipzig, 19. März 2017

Inhaltsverzeichnis

1	Einleitung	1
2	Ableitung der Schmerz Score	3
2.1	Segmentierung	3
2.2	Extrahierung von Eigenschaften und Ableitung der Schmerzscore	6
2.2.1	Extrahierung von Eigenschaften	8
2.2.2	Ableitung der Pain-Score	11
3	Visualisierung	16
4	Zusammenfassung	23
	Appendices	25
A	Auswertungsdetails VAD	26

Abbildungsverzeichnis

2.1	Beispiel für eine Segmentierung mit einem Grenzwert von $t_s = 6\text{ s}$	5
2.2	Beispiel für die Ableitung von Pain Scores für ein Signal nach einer fiktiven Pain Scale ohne Beobachtungszeitraum oder Aktualisierungsintervall	14
2.3	Beispiel für die Ableitung von Pain Scores für ein Signal nach einer fiktiven Pain Scale mit $t_{act} = 7.5\text{ s}$ und $t_{obs} = 15\text{ s}$	14
3.1	Oben: Ein Beispielsignal mit zwei Segmenten, die mit $t_s = 10\text{ s}$ gefunden wurden. Darunter: Pain Scores, die für die beiden Segmente berechnet wurden. Darunter: Schematische Darstellung des Signals als Regionen. Unten: Scores der Regionen.	17
3.2	Erstellte Farbpaletten mit zwei bis sieben Farben inklusive der zugehörigen RGB-Koordinaten im Hex-Code.	19
3.3	Visualisierung der „Length-Scale“ und der „Min-Scale“ für das Beispielsignal.	20
3.4	Oben: Beispielsignal mit zwei Segmenten. Unten: Überlappende Regionen, eingefärbt nach dem Farbschema der Min-Scale.	21
3.5	Visualisierung der Schmerzdiagnostik mit der Min-Scale bei Aktualisierungen. Die aktuellere Region wird <i>über</i> ältere Regionen gelegt.	22
3.6	Visualisierung der Schmerzdiagnostik mit der Min-Scale bei Aktualisierungen. Die aktuellere Region wird <i>unter</i> ältere Regionen gelegt.	22
A.1	Boxplot-Auswertung über Sensitivität, Spezifität und Genauigkeit der beiden besten Modelle	27

Tabellenverzeichnis

2.1	Fiktive Pain Scale	13
3.1	Fiktive Pain Scales zur Erläuterung der Visualisierung	16
3.2	Fixpunkte als Basis der Farbpaletten	19
3.3	Definition der Farbschemen zur Visualisierung der Pain Scales. Die Farbwerte werden als Hexadezimal-Codes für den RGB-Farbraum angegeben.	20
A.1	Klassifikationsgenauigkeiten der durch den REPTree entworfenen Modelle	26

1 Einleitung

Die Schmerzdiagnostik Neugeborener ist eine Aufgabe, die in vielen medizinischen Kontexten notwendig ist: Angefangen von der Überwachung prozeduraler Eingriffe wie Impfungen, über die Therapie von Entzugserscheinungen infolge des Drogenkonsums der Mutter bis hin zu Komplikationen bei der Geburt, welche die Beobachtung der Neugeborenen über mehrere Wochen bis Monate erfordert. Es gibt starke Hinweise darauf, dass insbesondere lang anhaltender Schmerz, welcher nicht korrekt erkannt und dessen Ursache somit nicht beseitigt wird, das Schmerzempfinden des Betroffenen bis in das Erwachsenenalter nachhaltig beeinflussen kann.[4, S. 402]

Babys äußern das Vorhandensein von Schmerz anhand einer Reihe von Merkmalen. Die für den Menschen am leichtesten sichtbaren Zeichen sind beispielsweise das Verkrampfen des Gesichtsausdruckes oder das Weinen. Aber auch physiologische Veränderungen wie der Erhöhung der Herzfrequenz oder des Blutdruckes lassen Rückschlüsse auf den Schmerzgrad zu.[9, S. 440] Auf Basis dieser Indikatoren werden im klinischen Alltag zur Schmerzdiagnostik sogenannte *multimodale Pain Scales* eingesetzt. Eine Pain Scale listet eine Reihe von Merkmalen, wie zum Beispiel den Hautfarbe vor das Weinen, für die eine Korrelation mit Schmerz festgestellt wurde. Für jedes Merkmal werden, je nach Grad der Ausprägung, Punkte vergeben, um schlussendlich den Schmerzgrad als *Pain Score* zu quantifizieren.[4, S. 406]

Forschungsbestrebungen der letzten Jahre haben sich zum Ziel gesetzt, die Schmerzbewertung zu automatisieren. Das Ziel dieser Arbeit ist die Entwicklung eines Konzeptes zur kontinuierlichen Schmerzbewertung bei Neugeborenen in einem multimodalen Verbund. Dabei soll der Schmerzgrad nicht nur abgeleitet, sondern auch visualisiert werden. Der Fokus liegt dabei auf der Analyse des Schmerzindikators *Weinen*, das heisst der Analyse akustischer Signale. Ausgangspunkt aller Entwürfe und Überlegungen sind die über viele Jahre erprobten Pain Scales.

Die Lösung dieser Aufgabenstellung erfordert die Kombination verschiedener Wissenschaftsdisziplin, wie zum Beispiel der medizinischen Schreiforschung oder der akustischen Modellierung der menschlichen Stimme. Kapitel ?? bietet eine Einführung in die wichtigsten Grundlagen. In Kapitel ?? wird ein Überblick über Veröffentlichungen mit ähnlichen Zielstellungen gegeben sowie dass in dieser Arbeit entwickelte Konzept grundlegend erläutert. Die folgenden Kapitel beleuchten die einzelnen Bausteine dieses Konzeptes genauer. In Kapitel ?? werden Methoden zur Feststellung des Vorhandenseins von Weingeräuschen in einem akustischen Signal vorgestellt. Dazu werden klassische Methoden der Voice Activity Detection bezüglich ihrer Leistungsfähigkeit im speziellen Anwendungsfall kindlicher Lautäußerungen eruiert und evaluiert. In Kapitel ?? werden Methoden entworfen, um aus den festgestellten Weingeräuschen den Schmerzgrad abzuleiten. Möglichkeiten der Visualisierung der Pain Score werden abschließend in Kapitel 3 vorgestellt.

Diese Arbeit leistet vor allem den Entwurf eines modular aufgebauten und erweiterbaren Konzeptes, welches die kontinuierliche Feststellung und Visualisierung von Schmerz Scores

ermöglicht. Das Modul, welches am ausgiebigsten evaluiert werden konnte, ist die Voice Activity Detection. Da alle darauf folgenden Module ausgiebigere Datenerhebungen in Zusammenarbeit mit medizinischen Fachkräften benötigen, wurden hier begründete Vorschläge zur Aufgabenbewältigung entworfen, welche jedoch nicht evaluiert werden konnten. Das Ziel dieser Arbeit war insbesondere nicht die Erlangung neuer Erkenntnisse über die tatsächliche Schmerzwahrnehmung Neugeborener.

2 Ableitung der Schmerz Score

In Kapitel ?? wurden Methoden vorgestellt, durch die einem Audiosignal die Cry-Units erkannt und markiert werden. Diese Cry-Units bilden den Ausgangspunkt, um eine Schmerz Score ableiten zu können.

Wie in Kapitel ?? erläutert wurde, wird eine Schmerz Score typischerweise nicht aus den Informationen einer einzelnen Cry-Unit, sondern aus Gruppen mehrerer Cry-Units innerhalb eines längeren Beobachtungszeitraumes geschlossen. Dafür ist es zunächst notwendig, diejenigen Cry-Units zu identifizieren, die gemeinsam einem schmerzversuchenden Stimulus zugeordnet werden können. Diese Gruppierung von Cry-Units wird in dieser Arbeit als *Segmentierung* bezeichnet und in Kapitel 2.1 vorgestellt. Die so entstandenen Segmente bilden die Basis zur Ableitung von Schmerz Scores, welche in Kapitel ?? diskutiert wird.

2.1 Segmentierung

Wie in Kapitel ?? erläutert wurde, wird die Schmerzdiagnose für Post-Operativen Schmerz mit Pain Scales typischerweise umgesetzt, indem das Baby in bestimmten Intervallen besucht und für einen festgelegten Zeitraum beobachtet wird, worauf den Schmerzgrad für einen bestimmten Zeitpunkt diagnostiziert wird. So empfiehlt die Pain Scale PAT, das Baby alle 30 Minuten zu besuchen und für 15 bis 30 Sekunden zu beobachten. Für viele Pain Scales konnten diese Beobachtungsintervalle und Zeiträume nicht in Erfahrung gebracht werden.

Dieses Vorgehen der Schmerzdiagnose nur zu festgelegten Zeitintervallen macht keinen Sinn für ein kontinuierliches System, da es den Vorteil der Kontinuität eliminieren würde. Die Frage ist also, wann eine medizinische Fachkraft den Anfangszeitpunkt einer Schmerzdiagnose festlegen würde, würde es das entsprechende Baby rund um die Uhr beobachten, und nicht nur eine Visite zu festgelegten Zeitpunkten durchführen. Die logisch nächstliegende Antwort ist, die Schmerzdiagnose genau dann zu starten, wenn das Baby anfängt, zu schreien.

Die nächste Frage ist, wann man die Beobachtung wieder beendet und eine Schmerz Score für den beobachteten Zeitraum festlegt. Da einige Pain Scales aus Tabelle ?? Beobachtungszeiträume angeben, kann argumentiert werden, dass die Beobachtung direkt nach dem für die jeweilige Scale angegebenen Zeitraum beendet werden und eine Pain Score abgeleitet werden kann, auch, wenn das Baby weiterhin schreit. Dabei stellt sich die Frage, welche Zeiträume man für die Pain Scales festlegt, die keinen selber keinen vorschreiben. Die zweite Frage ist, ob eine Vorzeitige Beendigung der Beobachtung überhaupt sinnvoll ist, wenn ein automatisierte System zur Verfügung steht und der Beobachter somit „unendlich viel Zeit hat“. Daher wird an dieser Stelle zuerst die grundlegendere Frage gestellt: Was ist der längste sinnvolle Zeitraum, für den eine Pain Score abgeleitet werden kann? Die nächstliegende Antwort ist: Dann, wenn das Kind aufhört, zu schreien.

Auf Basis dieser Argumentation wurde das folgende Vorgehen zur kontinuierlichen Seg-

mentierung entwickelt: Wenn das Baby keine Äußerungen von sich gibt, weil es beispielsweise schläft, wird keine Cry-Unit festgestellt, und somit existiert auch momentan kein offenes Segment. Fängt das Baby an, einen Laut von sich zu geben, also eine Cry-Unit zu produzieren, wird ein neues Segment eröffnet und die Cry-Unit diesem Segment hinzugefügt. Weitere Cry-Units werden so lange diesem Segment hinzugefügt, wie die Dauer der Stille nach einer Cry-Unit einen festgelegten Grenzwert t_s nicht überschreitet. Ein Cry-Segment wird dann geschlossen, wenn das Baby für einen festgelegten Zeitraum keine Laute mehr von sich gibt, also „aufhört, zu weinen“. Das Endzeitpunkt des Segmentes wird als der Endzeitpunkt der letzten Cry-Unit des Segmentes festgelegt.

Formel 2.1 definiert ein *Cry-Segment* $[CS]$ als Datentyp. Ein Cry-Segment ist eine Liste von Cry-Units. Alle Cry-Units erfüllen die Nebenbedingung 2.2, das heißt, dass die Distanzen aller benachbarter Cry-Units eines Cry-Segments unterhalb des Grenzwertes t_s liegen.

$$CS = [cu_0, \dots, cu_n] \quad (2.1)$$

$$\forall cs \in CS : \forall i = 0 \dots \text{length}(cs) - 2 : d(cs[i], cs[i + 1]) < t_s \quad (2.2)$$

Der Start-Zeitpunkt eines Cry-Segments wird nach Formel 2.3 als der Startzeitpunkt der ersten Cry-Unit des Segments definiert. Das Ende eines Segmentes wird definiert als der Endzeitpunkt der letzten Cry-Unit nach Gleichung 2.4.

$$\text{start}(cs) = cs[0].\text{start} \quad (2.3)$$

$$\text{end}(cs) = cs[n].\text{end} \quad (2.4)$$

Algorithmus 1 zeigt einen Pseudocode, wie die Segmentierung nach dem beschriebenen Prinzipien offline durchgeführt wird. Input des Algorithmus ist die Liste aller Cry-Units $CU_{all} = [cu_0, \dots, cu_m]$, die durch das Decision-Smoothing nach Algorithmus ?? entstanden ist. Das Ergebnis des Algorithmus ist die Liste CS_{all} , die alle gefundene Cry-Segmente $[cs_0, \dots, cs_n]$ enthält. Der Algorithmus eignet sich nicht für eine Online-Segmentierung, da das Ende eines Segmentes erst bei Beginn eines neuen Segmentes festgestellt wird, wobei beliebig viel Zeit zwischen den Beiden Segmenten liegen kann. Wurde beispielsweise $t_s = 1 \text{ min}$ festgelegt, und die Pause zwischen zwei Segmenten beträgt eine Stunden, so wäre das Ende des ersten Segmentes 59 Minuten zu spät festgestellt worden. Bei einer online durchgeführten Segmentierung empfiehlt es sich, die Dauer der Stille nach jeder neu erkannten Cry-Unit kontinuierlich zu messen und ein Segment sofort zu beenden, wenn ein Stillezeitraum den Grenzwert t_s überschreitet.

Abbildung 2.1 zeigt die nach dieser Methode durchgeführte Segmentierung anhand eines Beispiels.

Das hier vorgestellte Vorgehen wurde absichtlich möglichst einfach gehalten, damit der Sinn des Parameters t_s leicht ersichtlich ist und somit von der medizinischen Fachkraft selbstständig festgelegt werden kann. Schlussendlich ist eines der Hauptziele dieser Segmentierung, unnötige Berechnungen von Schmerz-Scores in den nachfolgenden Schritten zu vermeiden, so lange keine Cry-Units vorliegen. Das Ende eines Segmentes ist außerdem ein günstiger Zeitpunkt, um die Parameter des Kompressors im Pre-Processing auf Basis des

Algorithm 1 Gruppierung von Cry-Units zu Cry-Segments

```

1: function SEGMENTCRYUNITS( $CU_{all}, t_s$ )
2:    $CS_{all} \leftarrow []$ 
3:    $cs \leftarrow [CU_{all}[0]]$ 
4:   for  $i = 1, \dots, \text{length}(CU_{all}) - 1$  do
5:      $cu_i \leftarrow CU_{all}[i]$ 
6:      $cu_{i-1} \leftarrow CU_{all}[i - 1]$ 
7:     if  $d(cu_{i-1}, cu_i) < t_{seg-max}$  then
8:        $cs \leftarrow [cs_i, cu_i]$ 
9:     else
10:       $CS_{all} \leftarrow [CS_{all}, cs]$ 
11:       $cs \leftarrow [cu_i]$ 
12:    end if
13:  end for return  $CS_{all}$ 
14: end function

```

RMS-Wertes des Segmentes zu aktualisieren (siehe Kapitel ??). Trotz der Trivialität dieser laufenden Segmentierung liegt hier ein wichtiger Unterschied im Gegensatz zu vergleichbaren Systemen, wie zum Beispiel das von Cohen et al. [2], bei dem die Entscheidung über Cry/not-Cry für Segmente mit einer festen Fenstergröße von 10 Sekunden vorgenommen wird.

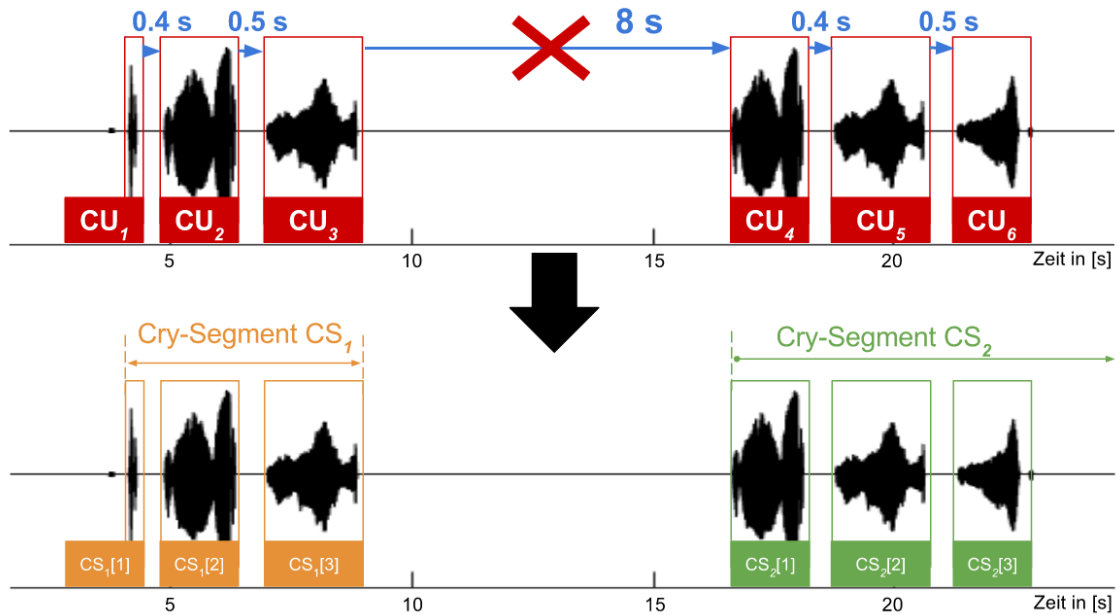


Abbildung 2.1: Beispiel für eine Segmentierung mit einem Grenzwert von $t_s = 6$ s

2.2 Extrahierung von Eigenschaften und Ableitung der Schmerzscore

Das Ergebnis der Segmentierung ist eine Liste an Cry-Segmenten cs_0, \dots, c_n . Diese Cry-Segmente bilden nun die Basis für die Ableitung der Pain Score. Die medizinische Fachkraft, die das System verwendet, muss dabei zuerst die Wahl treffen, welche Pain Scale verwendet werden soll. Das einfachste denkbare Vorgehen ist die Ableitung genau einer Pain Score aus den globalen Eigenschaften eines Segmentes, wobei diese Ableitung erst vollzogen werden kann, sobald ein Segment abgeschlossen wurde und alle Informationen für dieses Segment vorliegen. Es wird also jedem Segment genau eine Pain Score zugewiesen. Das Vorgehen wird am Beispiel der NIPS aus Tabelle ?? verdeutlicht: Dabei steht die Abwesenheit von Weinen für null Punkte, „mumbling“ (murmeln) für einen Punkt und „vigorous“ (energisch) für zwei Punkte. Bei Abwesenheit von Lautäußerungen, also der Zeitraum zwischen den Segmenten, werden also keine Punkte = null Punkte vergeben. Ein Segment, dessen Qualität insgesamt als „murmelnd“ bewertet wird, erhält einen Punkt, und ein Segment, welches als insgesamt als „energisch“ bewertet wird, zwei Punkte. Das Problem ist offensichtlich: „murmelnd“ und „energisch“ sind subjektiv behaftete Begriffe und lassen sich nicht ohne weiteres aus den Eigenschaften eines Segmentes feststellen.

Es werden zwei verschiedene Lösungs-Strategien für dieses Problem vorgestellt.

Strategie 1

... löst das Problem mit Hilfe von *Regression* (Siehe Kapitel ??):

1. Man erstellt eine Datenbank mit Aufnahmen von kindlichen Lautäußerungen, die man segmentiert.
2. Man errechnet „so viele *objektiv* messbare Eigenschaften wie möglich“ für jedes Segment, wie zum Beispiel die insgesamt Länge, die durchschnittliche Länge der enthaltenen Cry-Units, durchschnittliche Tonhöhe usw.
3. Man bittet medizinische Fachkräfte, für jedes Segment der Datenbank eine Score bezüglich einer Pain Scale zu vergeben. Dadurch erhält man eine gelabelte Test-Datenbank.
4. Man verwendet einen *Regressionsalgorithmus*, um den Zusammenhang zwischen den in Schritt 2 objektiv gemessenen Eigenschaften der Segmente und den in Schritt 3 vergebenen *Scores* herzustellen. An dieser Stelle kann zum Beispiel die in Kapitel ?? beschriebene multiple lineare Regression verwendet werden. Man erhält somit einen Regressor für jede Pain Scale.
5. Möchte man für neue, unbekannte Segmente die Pain Score prognostizieren, nutzt man den entsprechenden Regressor.

Das Vorteil dieses Vorgehens ist, dass das Problem der Übersetzung der objektiv messbaren Parameter in die subjektiv behafteten Begriffe überbrückt wird, indem die Regression direkt von den objektiv messbaren Parametern auf die Pain Score durchgeführt wird. Der Nachteil ist, dass eine Testdatenbank für jede Pain Scale aufgebaut werden muss. Wird ein neue Pain Scale eingeführt, muss der Regressor für diese Scale durch erneutes Labeln festgestellt werden. Ein weiterer Effekt der Abbildung des Problems als Regression ist, dass ein Regressor

in einen kontinuierlichen Zahlenraum abbildet. Es sind also Regressionsergebnisse wie zum Beispiel 2.8 denkbar. Diese „bessere Auflösung“ kann als Vorteil betrachtet werden. Ist jedoch eine direkte Übersetzung der Pain Scale inklusive der ganzzahligen Punktzahlen gewünscht, so stellt sich die Frage, ob eine 2.8 auf- oder abzurunden ist.

Strategie 2

... löst das Problem mit Hilfe von Klassifizierung (Siehe Kapitel ??):

1. und 2. entsprechen Strategie 1
3. Man sammelt alle subjektiven Begriffe, die in Pain Scales verwendet werden, wie zum Beispiel „murmelnd“, „energisch“, usw.
4. Man bittet medizinische Fachkräfte, jedes Segment der Datenbank mit denjenigen Begriffen zu labeln, die die jeweilige Person für zutreffend hält.
5. Man Verwendet einen *Klassifizierungsalgorithmus*, um einen Zusammenhang zwischen den in Schritt 2 festgestellten objektiv messbaren Eigenschaften der Segmente und den *subjektiv behafteten Begriffen* zu finden. Man erhält somit einen Klassifikator für jeden Begriff, der binär in *positive = zutreffend* und *negative = nicht zutreffend* klassifiziert.
6. Möchte man für neue, unbekannte Segmente die Pain Score prognostizieren, so wird für jede mögliche Score der Pain Scale überprüft, ob für alle subjektiv beschreibenden Begriffe der entsprechende Klassifikator ein positive prognostiziert. Die Ableitung der Score ist somit ein weiteres Klassifizierungsproblem, wobei eine Score einer Klasse entspricht und genau dann abgeleitet werden kann, wenn alle Voraussetzungen für die Klasse erfüllt sind.

Der Vorteil dieser Methode ist, dass auch zum Zeitpunkt der Erstellung der Testdatenbank unbekannte Pain Scales zu einem späteren Zeitpunkt eingebunden werden können, insofern alle in dieser neuen Pain Scale verwendeten subjektiv behafteten Begriffe bereits gelabelt vorliegen, weil sie auch in anderen Pain Scales verwendet wurden. Das Vorgehen erlaubt somit eine gewissen Flexibilität bezüglich zukünftig entwickelter Pain Scales. Der Nachteil dieser Methode ist, dass durch die Umwandlung der eigentlich quantitativ geordneten Score einer Pain Scale in qualitative Klassen aus einem implizit als Regression zu betrachtenden Problem ein Klassifizierungsproblem macht. Dies wirft neue Fragen auf, wie zum Beispiel: Angenommen, bei einer fiktiven Pain Scale wird jede Score mit jeweils drei subjektiv behafteten Begriffen beschrieben, und bei der Klassifizierung eines Segmentes wird festgestellt, dass für jede Punktzahl genau zwei der drei Begriffe erfüllt werden. Welche Score wird dann prognostiziert? Ein anderes Beispiel wird am Beispiel der der NIPS-Score aus Tabelle ?? verdeutlicht: Angenommen, ein Cry-Segment enthält hörbar „starkes“ Schreien, es kann jedoch weder „mumbling (murmelnd)“ noch „vigorous (energisch)“ abgeleitet werden. Demzufolge müsste dieses Segment eine Score von 0 Punkten erhalten, wobei ein Mensch in dieser Situation eventuell „stark“ zu „heftig“ uminterpretieren und 2 Punkte vergeben hätte. Strategie 1 ist weniger anfällig für dieses Problem.

In jedem Fall werden medizinische Fachkräfte benötigt, um das Labeling der Cry-Segmente durchzuführen, was aus Zeitgründen im Rahmen dieser Arbeit nicht möglich ist. Die Aquisierung von Audioaufnahmen von Babys sowie das Labeling der Aufnahmen erfordern nicht nur Zeit, sondern das Fachwissen über das Führen und die Auswertung von Interviews.

2.2.1 Extrahierung von Eigenschaften

Im vergangenen Kapitel wurde erläutert, dass die Basis für die Ableitung einer Pain Score für ein Segment die Extraktion von „so vielen Features wie möglich“ ist. In diesem Kapitel wird präzisiert, welche Features gemeint sind. Varallyay [11, S. 16 - 17] schlug vor, drei Kategorien an Features zu betrachten: 1.) Features des Zeitbereichs, 2.) Features der Frequenzbereichs, und 3.) Melodie-bezogene Attribute. Diese Kategorisierung wurde für diese Arbeit übernommen.

In Kapitel ?? wurde beschrieben, welche Features in der medizinischen Schreiforschung typischerweise extrahiert wurden. In Kapitel ?? wurde diskutiert, dass 1.) nicht bewiesen ist, welche Features die „wichtigsten“ sind und 2.) keine Einigung darüber herrscht, wie genau bestimmte Features zu berechnen sind. Basierend auf den in diesem Kapitel vorgestellten Features werden in diesem Kapitel konkrete Berechnungsvorschriften definiert. Welche von diesen Features tatsächlich im Zusammenhang mit Schmerz stehen, lässt sich erst in der anschließenden Nutzung der Features zur Regression oder Klassifizierung der Pain Scales feststellen, welche jedoch im Rahmen dieser Arbeit nicht durchgeführt werden kann.

Features des Zeitbereiches

Mit Features des Zeitbereiches sind solche gemeint, die sich allein aus Kenntnis der der Start- und Endzeitpunkte der im Segment enthaltenen Cry Units sowie deren Zeitbereiche gewinnen lassen, wie beispielsweise die durchschnittliche Länge der Cry-Units oder die durchschnittliche Energie der Cry-Units. In diesem Kapitel gilt die Konvention, dass eine Cry-Segment cs insgesamt N Cry-Units enthält, die Indexierung wird mit $0 \dots N - 1$ definiert.

Segment-Length: Zeitliche Länge des Segmentes:

$$\text{Segment-Length}(cs) = cs[N - 1].end - cs[0].start \quad (2.5)$$

Density: Relativer Anteil der Cry-Units an der Länge des Segmentes („Dichte“)

$$\text{Density}(cs) = \frac{\sum_{i=0}^{N-1} \lambda(cs[i])}{\text{Segment-Length}(cs)} \quad (2.6)$$

Tempo: Das Verhältnis zwischen der Dauer des Segmentes und der Anzahl der Cry-Units. Dieses Feature ähnelt dem von LaGasse et al. [6, S. 85] als *Utterances* bezeichneten Feature.

$$\text{Tempo}(cs) = \frac{N}{\text{Segment-Length}(cs)} \quad (2.7)$$

Statistics of Cry-Units: Statistische Auswertungen bezüglich der *Länge der Cry-Units* $\text{stats}_{cu}(cs)$: Durchschnitt, Median, Minimum, Maximum und Standardabweichung

der Cry-Units. Das $\text{mean}_{cu}(cs)$ -Feature wird von LaGasse et al. [6, S. 85] und vielen weiteren Schreiforschern als *Mean Duration* bezeichnet.

$$\text{stats}_{cu}(cs) = \begin{cases} \text{mean}_{cu}(cs) = \text{mean}_{i=0 \dots N-1} \{\lambda(cs[i])\} \\ \text{median}_{cu}(cs) = \text{median}_{i=0 \dots N-1} \{\lambda(cs[i])\} \\ \text{min}_{cu}(cs) = \min_{i=0 \dots N-1} \{\lambda(cs[i])\} \\ \text{max}_{cu}(cs) = \max_{i=0 \dots N-1} \{\lambda(cs[i])\} \\ \sigma_{cu}(cs) = \sigma_{i=0 \dots N-1} \{\lambda(cs[i])\} \end{cases} \quad (2.8)$$

Statistics of Bursts: ¹ Die in Gleichung 2.8 definierten Features können ebenso in Bezug auf die *Längen der Bursts* errechnet werden, in dem in jeder Gleichung $\lambda(cs[i])$ ersetzt wird durch $cs[i].start - cs[i-1].start$. Die Indexierung muss auf $i = 1, \dots, N-1$ begrenzt werden.

$$\text{stats}_{burst}(cs) = \begin{cases} \text{mean}_{burst}(cs) = \text{mean}_{i=1 \dots N-1} \{cs[i].start - cs[i-1].start\} \\ \text{median}_{burst}(cs) = \text{median}_{i=1 \dots N-1} \{cs[i].start - cs[i-1].start\} \\ \dots \end{cases} \quad (2.9)$$

Statistics of Pauses: Nach dem selben Muster werden die statistischen Auswertungen bezüglich der *Längen der Pausen* ermittelt. Eine Pause entspricht in diesem Zusammenhang der Distanz zwischen zwei aufeinander folgenden Cry-Units nach Gleichung ??.

$$\text{stats}_{pause}(cs) = \begin{cases} \text{mean}_{pause}(cs) = \text{mean}_{i=1 \dots N-1} \{d(cs[i-1], cs[i])\} \\ \text{median}_{pause}(cs) = \dots \end{cases} \quad (2.10)$$

Statistics of Energies: Zunächst wird die Liste aller in den Cry-Units enthaltenen Signalfenster definiert nach Gleichung 2.11. Eine Cry-Unit hat die Signalfenster $cu.windows = x_0[], \dots, x_m[]$

$$x_{seg}[] = cs[0].windows[0], \dots, cs[N-1].windows[m] \quad (2.11)$$

Die Liste $x_{seg}[]$ hat R Elemente, die Indexierung wird definiert mit $0, \dots, R-1$. Gleichung 2.12 definiert die Features bezüglich der MSV-Werte („Lautstärken“) des Segmentes. Der MSV-Wert als Maß des durchschnittlichen Energiegehaltes wurde in Gleichung ?? definiert.

$$\text{stats}_{msv}(cs) = \begin{cases} \text{mean}_{msv}(cs) = \text{mean}_{i=0 \dots R-1} \{MSV(x_{seg}[i])\} \\ \text{median}_{msv}(cs) = \dots \end{cases} \quad (2.12)$$

¹Erläuterung zum Begriff *Burst* in Kapitel ??

Diese statistischen Auswertungen bezüglich der Länge der Cry-Units und Bursts wurden beispielsweise von Zeskind et al. [10] vorgenommen, wenn auch nicht Computer-gestützt. Es ist zu bemerken, dass in der klassischen Schreiforschung zeitliche Features im geringeren Maße in Betracht gezogen wurden als Features des Frequenz-Bereiches. Die einzigen zeitliche Features, die zum Beispiel von Wasz-Hockert et al. [8], Fuller [3] und LaGasse et al. [6] berechneten, sind *die durchschnittliche Länge der Cry-Units* (hier $\text{mean}_{cu}(cs)$) und die *Latenz zwischen Reiz und erster Cry-Unit*, welche nur auf Basis des Audiosignals nicht feststellbar ist. Es spricht jedoch nichts dagegen, die hier vorgestellten Features trotzdem zu erproben. Die anschließende Nutzung der Features zur Regression/Klassifizierung wird Auskunft darüber geben, welchen Beitrag diese Features zur Schmerzdiagnose leisten können.

Features des Frequenzbereiches und der Melodie

Mit Features des Frequenz-Bereiches sind diejenigen Features gemeint, die sich aus der Short Time Fourier Transformation der Cry-Units gewinnen lassen. Um die Features durch mathematische Formeln definieren zu können, wird zuerst das *Spectrum des Segmentes* $X_{seg}[\]$ nach Formel 2.13 als die Liste aller Frequenz-Bereiche der Signalfenster der Cry-Units des Segmentes definiert. Die Indexierung von $X_{seg}[\]$ läuft, wie bei $x_{seg}[\]$ von $0, \dots, R-1$. Nach dem selben Muster wird das *Cepstrum des Segmentes* $c_{seg}[\]$ definiert.

$$X_{seg}[\] := \bigvee_{x_i[\] \in x_{seg}} : |DFT\{x_i[\] \cdot w[\]\}| \quad (2.13)$$

Die folgenden Features des Frequenzbereiches lassen sich mit den in dieser Arbeit vorgestellten Methoden berechnen:

Tensness: Das Feature, welches in Kapitel ?? als „Ratio2“ beschrieben wurde. Es wurde von Fuller [3] eingeführt und beschreibt die Spannung des Vokaltraktes als Verhältnis der Energien oberhalb von 2000 Hz zu unter 2000 Hz. Wie bei den statistischen Auswertungen der Features des Zeitbereiches kann für das gesamte Segment der Durchschnitt, Median, Maximum, Minimum und Standardabweichung berechnet werden.

$$\text{stats}(Tensness) = \begin{cases} \text{mean}_{Tens}(cs) = \text{mean}_{i=0 \dots R-1} \left\{ \frac{\sum_{k=0}^{2000 \text{ Hz}} X_{sec}[i][k]}{\sum_{j=2000 \text{ Hz}}^{f_s} X_{sec}[i][j]} \right\} \\ \text{median}_{Tens}(cs) = \dots \end{cases} \quad (2.14)$$

Clarity: Wie in Kapitel ?? erläutert wurde, lässt eine stark ausgebildete Spitze im oberen Cepstrum-Bereich auf ein stimmhaftes Signal schließen. Ein hoher Anteil stärkerer Cepstrum-Peaks lässt also auf vermehrt phonierte Laute schließen, geringere Cepstrum-Peaks auf dysphoniertere Laute (Siehe Kapitel ??). Dieses durchschnittliche Wert dieses Features trifft eine Aussagen über den Anteil dysphonierter Laute, die Standardabweichung ähnelt dem in Kapitel ?? vorgestellten *Cry-Mode Changes*-Feature.

$$\text{stats}_{clarity}(cs) = \begin{cases} \text{mean}_{Clarity}(cs) = \text{mean}_{i=0 \dots R-1} \left\{ Ceps_{mag}(c_{seg}[i]) \right\} \\ \text{median}_{Clarity}(cs) = \dots \end{cases} \quad (2.15)$$

Alle weiteren Features, die in Kapitel ?? vorgestellt wurden und sich auf den Frequenzbereich beziehen, lassen sich nicht mehr mit den in dieser Arbeit vorgestellten Methoden extrahieren. Entweder beziehen sie sich auf die Lage der Formanten, oder basieren auf der Feststellung der Grundtonhöhe. In dieser Arbeit konnten aus Platzgründen jedoch keine Methoden zur Extraktion dieser Informationen vorgestellt werden. Gleiches gilt für die Feststellung des Melodieverlaufs, welche ebenfalls auf der Feststellung der Grundtonhöhe basiert. Das Muster, nach dem diese Features berechnet werden können, sollte aus den bisher vorgestellten Features ersichtlich sein. So lassen sich beispielsweise die Features bezüglich der Grundtonhöhe nach Formel 2.16 ableiten. Dabei sei $f_0(X_i[])$ eine idealisierte Funktion, welche die Grundtonhöhe f_0 für das Frequenzfenster $X_i[]$ berechnet. Da für die Definition der weiteren Features idealisierte ebenfalls Funktionen angenommen werden müssten, wird die Festlegung weiterer Features an dieser Stelle nicht fortgeführt.

$$\text{stats}_{pitch}(cs) = \begin{cases} \text{mean}_{Pitch}(cs) = \text{mean}_{i=0 \dots R-1} \left\{ f_0(X_{seg}[i]) \right\} \\ \text{median}_{Pitch}(cs) = \dots \end{cases} \quad (2.16)$$

Diskussion

Bei allen vorgestellten Features handelt es sich, nach dem Vorbild der in Kapitel ?? vorgestellten Features der klassischen Schreiforschung, um solche, bei denen die Reihenfolge der Cry-Units nicht mit in Betracht gezogen wird. Angenommen, ein Segment besteht aus n Cry-Units, wobei genau eine Hälfte der Cry-Units kurz und die andere Hälfte der Cry-Units lang ist. Das $\text{stats}_{cu}(cs)$ -Feature wird bezüglich des Durchschnittes, Minimum, Maximum etc. die selben Werte berechnen, unabhängig davon, ob sich die kurzen Cry-Units allesamt am Beginn des Segmentes, am Ende des Segmentes oder mit den langen Cry-Units durchmischt befinden. Bei der anschließenden Nutzung der Features zu Regression/Klassifizierung wird sich zeigen, wie sehr sich diese Features zur Ableitung von Pain-Scores eignen. Stellt sich heraus, dass sich die Features nicht eignen, ist es eventuell notwendig, neue Features zu definieren, die die Position der Cry-Units mit in Betracht ziehen.

2.2.2 Ableitung der Pain-Score

Mathematisch wird die Ableitung einer Pain Score als Funktion $PS_{Scale} : CS \mapsto S_{Scale}$ definiert, welche eine Cry-Segment $cs \in CS$ unter Verwendung der Pain-Scale auf einen Pain Score $s \in S_{Scale}$ abbildet. Die Menge der Pain Scores S_{Scale} ist abhängig von der verwendeten Pain Scale. Es ist zu beachten, dass selbst, wenn zwei Pain Scales einen Score von 2 enthalten, diese 2 *nicht* zwingendermaßen auf den selben Schmerzgrad hinweist. Der konkrete Inhalt der Funktion ist für die entsprechende Pain Scale mit dem in diesem Kapitel beschriebenen Vorgehen zu ermitteln. Die Ableitung genau einer Pain Score für ein Cry-Segment stellt den einfachsten Fall dar. Dies ist für bestimmte Anwendungsfälle eventuell nicht ausreichend:

1. Die Score kann erst nach der Beendigung eines Segmentes abgeleitet werden, was für einigen Kontexte möglicherweise zu spät ist. Besonders die Schmerzdiagnostik während Schmerzverursachenden Prozeduren kann das häufigere „Aktualisieren“ der Schmerzscores notwendig machen.
2. Falls der Schmerz innerhalb eines Segmentes stark ab- oder zunimmt, ist dieser Verlauf

nicht erkennbar. Es würde lediglich der „durchschnittliche Schmerz“ des Segmentes abgeleitet werden.

Das vorgestellte Prinzip wird daher erweitert, indem ein Aktualisierungsintervall t_{act} und Beobachtungszeitraume t_{obs} eingeführt wird.

Aktualisierungsintervall

Die Grundlegende Idee des Aktualisierungsintervalls ist, bei einem momentan offenen Segment in regelmäßigen Abständen die Features abzufragen und direkt die Pain Score abzuleiten, um Zwischenergebnisse zu erhalten. Der am häufigsten umsetzbare Fall ist, ein Aktualisierung nach jeder neu dem Segment hinzugefügten Cry-Unit vorzunehmen. Der am wenigsten häufige Fall ist der bereits genannte, die Aktualisierung erst bei Beendigung eines Segmentes durchzuführen. Eine offensichtliche Variante zur Festlegung von t_{act} ist die Bestimmung eines zeitlichen Wertes. Ein t_{act} von beispielsweise 10s würden bedeuten, dass alle 10 Sekunden ein neuer Pain Score für ein Segment berechnet wird. Die Beendigung eines Segmentes würde in jedem Fall eine Ableitung der Pain Score auslösen und einen „erzwungenen Aktualisierungszeitpunkt“ darstellen. Die folgenden Möglichkeiten zur Festlegung von t_{act} sind denkbar:

- t_{act} als globaler, nicht veränderbarer Wert. Da in der Literatur keine Vorschlag diesbezüglich gefunden werden konnte, müsste ein sinnvoller Wert in Absprache mit medizinischen Fachkräften eruiert werden.
- Man überlässt der medizinischen Fachkraft, die das System überwacht, die Festlegung des Aktualisierungsintervalls. So kann die Person selber bestimmen, wie häufig sie eine Aktualisierung der Pain Score wünscht.
- Die feste Bindung des Aktualisierungsintervalls an die verwendete Pain Scale. Die CRIES-Scale ist beispielsweise für das post-operative Monitoring gedacht und benötigt somit möglicherweise weniger häufige Aktualisierungen als der DAN, welcher zur Schmerzdiagnostik während einer Operation eingesetzt werden kann (siehe Tabelle ??). Da die Pain Scales nicht für die kontinuierliche Schmerzdiagnostik ausgelegt sind, lässt sich aus den für einige Pain Scales eventuell vorgeschriebenen Beobachtungsintervallen kein Aktualisierungsintervall für ein kontinuierliches System argumentieren.

Wenn t_{act} als zeitlicher Wert definiert wird, kann es passieren, dass eine Aktualisierung in einem offenen Segment durchgeführt wird, während gerade eine neue Cry-Unit markiert wird und noch nicht abgeschlossen wurde. Da die Funktion $PS_{Scale} : cs \mapsto \mathbb{N}$ nur für Cry-Segmente definiert wurde, die vollständige Cry-Units enthalten, wird diese „halbe Cry-Unit“ nicht mit zur Ableitung der Pain Score verwendet. Der Hintergrund für diese Entscheidung ist, dass bestimmte Funktionen zum Ableiten der Features für das Segment ansonsten fehlerhafte Ergebnisse liefern können. Die Funktion PS wird folglich auf das *Subsegment* cs_{sub} angewandt, welches alle Cry-Units des möglicherweise noch offenen Segmentes cs beinhaltet, die zum Aktualisierungszeitpunkt t vollständig begonnen und beendet wurden. Der Endzeitpunkt des Subsegmentes wird, wie bei herkömmlichen Segmenten, auf den Endzeitpunkt der letzten vollständigen Cry-Unit im Subsegment gelegt.

Beobachtungszeitraum

Es gibt Eigenschaften, die sich implizit auf den gesamten Zeitraum *Beginn des Segmentes* $start(cs)$ bis *Aktualisierungszeitpunkt* t beziehen, wie beispielsweise die *Zeitliche Länge des Segmentes* nach Formel 2.5. Dieser Zeitraum ist gleichzeitig der längst mögliche Zeitraum innerhalb eines Segmentes, der für die Ableitung der Pain Score mit einbezogen werden kann. Es ist jedoch auch möglich, einen kürzere Beobachtungszeitraum t_{obs} zu wählen. Dies hat zur Folge, dass bei der Ableitung die ersten Cry-Units des Segmentes ausgelassen werden, die außerhalb des Beobachtungszeitraums liegen. So können zeitliche Veränderungen der Pain-Score innerhalb eines Segmentes detaillierter dargestellt werden. Es sind wiederum verschiedene Varianten zur Festlegung von t_{obs} denkbar:

- Festlegung eines globalen oder eines von der medizinischen Fachkraft frei wählbaren Wertes, so wie bei dem Aktualisierungsintervall t_{act} .
- Die feste Bindung des Aktualisierungsintervalls an die verwendete Pain Scale. Einige Pain Scale empfehlen bestimmte Beobachtungszeiträume. So wird beispielsweise bei der NIPS-Scale ein Beobachtungszeitraum von einer Minute empfohlen (Siehe Kapitel ??). Es müsste wiederum in Zusammenarbeit mit medizinischen Fachkräften eruiert werden, ob diese, für die manuelle Schmerzdiagnostik vorgesehenen Werte auch für ein automatisiertes System Sinn machen.
- Eine weitere Variante ist, t_{obs} an den Wert von t_{act} zu binden. Fall t_{obs} frei festlegbar sein soll, muss das Personal nicht zwei Werte festlegen. Ein Verhältnis von $t_{obs} = k \cdot t_{act}$ würde mit $k = 1$ nicht-überlappende Beobachtungszeiträume und mit $k = 2$ überlappende Beobachtungszeiträume erzeugen.

Der Beobachtungszeitraum t_{obs} schränkt somit die Länge des Subsegmentes cs_{sub} weiter ein, und zwar in diesem Fall bezüglich der Startzeitpunktes. Der Zeitraum innerhalb des Segmentes cs , der zur Bildung von cs_{sub} genutzt wird, ist der Zeitraum *Aktualisierungszeitpunkt* $t - t_{obs}$ bis t . Es werden nur solche Cry-Units von cs in cs_{sub} übernommen, die innerhalb dieses Zeitraumes vollständig begonnen und beendet werden konnten. Der Anfangszeitpunkt des Subsegments ist somit der Anfangszeitpunkt der ersten vollständigen Cry-Unit innerhalb Beobachtungszeitraumes, der Endzeitpunkt des Subsegmentes entspricht dem Ende der zum Aktualisierungszeitpunkt zuletzt vollständig beendeten Cry-Unit.

Beispiel

Die in diesem Kapitel vorgestellten Methoden zur Ableitung von Pain Scores werden anhand eines Beispiels verdeutlicht. Tabelle 2.1 definiert eine fiktive Pain Scale. Es werden die für Pain Scales typischen, subjektiv behafteten Worte zum Scoring verwendet.

Tabelle 2.1: Fiktive Pain Scale

	0 Punkte	1 Punkt	2 Punkte
Fiction Scale	kein oder sehr wenig Weinen	normales Weinen	starkes Weinen

Mit Hilfe der in Kapitel 2 beschriebenen Strategien wurde die Funktion $PS_{Fiction} : CS \mapsto$

$\{0, 1, 2\}$ ermittelt, definiert in Gleichung 2.17. Sie erlaubt die Ableitung des Schmerz Score für eine Cry-Segment mit Hilfe objektiv messbarer Features.

$$PS_{Fiction}(cs) = \begin{cases} 0 & , \text{wenn } mean_{cu}(cs) < 0.3 \text{ s} \\ 1 & , \text{wenn } mean_{cu}(cs) < 1 \text{ s} \\ 2 & , \text{wenn } mean_{cu}(cs) \geq 1 \text{ s} \end{cases} \quad (2.17)$$

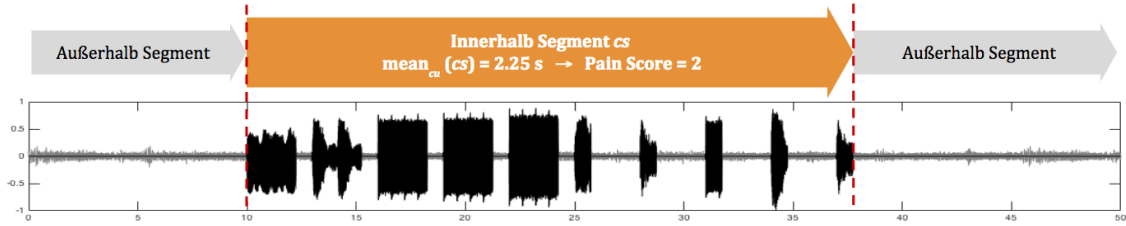


Abbildung 2.2: Beispiel für die Ableitung von Pain Scores für ein Signal nach einer fiktiven Pain Scale ohne Beobachtungszeitraum oder Aktualisierungsintervall

Abbildung ?? zeigt ein Beispielsignal, für das Pain Scores nach dieser Pain Scale abgeleitet werden. In dem Signal werden die stimmhaften Signalbereiche schwarz und das Hintergrundrauschen grau dargestellt. Es sind insgesamt 10 Cry-Units zu erkennen. Die ersten fünf Cry-Units haben jeweils eine Länge von 2.25 s, die letzten fünf Cry-Units eine jeweilige Länge von 0.75 s. Das Signal wurde nach der in Kapitel 2.1 beschriebenen Methode segmentiert mit $t_s = 5 \text{ s}$ und so alle 10 Cry-Units zu einem Segment zusammengefasst. Das Segment erstreckt sich von Sekunde 10 bis Sekunde 37.5. Für das Segment wurde eine durchschnittliche Länge der Cry-Units von $mean_{cu}(cs) = 1.5 \text{ s}$ gemessen und dem zufolge eine Pain Score von 2 abgeleitet. In diesem Fall wurde ohne Beobachtungs- und Aktualisierungsintervall gearbeitet. Wäre die Analyse also kontinuierlich vorgenommen worden, so wäre nach Feststellung der ersten Cry-Unit das Segment eröffnet, nach Überschreitung der maximal zulässigen Stille von $t_s = 5 \text{ s}$ nach der 10. Cry-Unit das Segment geschlossen, und daraufhin $PS_{Fiction}(cs) = 2$ berechnet worden.

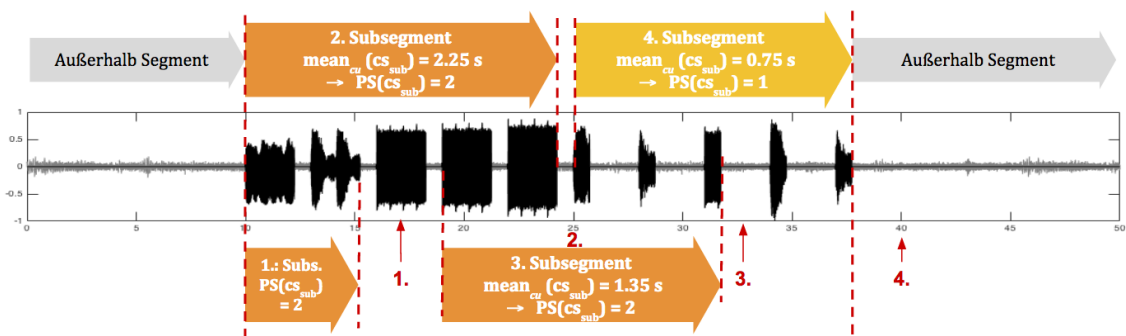


Abbildung 2.3: Beispiel für die Ableitung von Pain Scores für ein Signal nach einer fiktiven Pain Scale mit $t_{act} = 7.5 \text{ s}$ und $t_{obs} = 15 \text{ s}$

Abbildung ?? zeigt die Ableitung der Schmerz Scores, wenn zusätzlich ein Aktualisierungsintervall von 7.5 s und ein Beobachtungszeitraum von 15 s gewählt wird. Nach dem das Segment durch die Cry-Unit an Sekunde 10 eröffnet wurde, werden Aktualisierungen zu den Zeitpunkten $t = 17.5 \text{ s}, 25 \text{ s}, 32.5 \text{ s}$ und 40 s durchgeführt, verdeutlicht durch die

kleinen, roten Pfeile in der Abbildung. Wie zu sehen ist, wird bei jeder Aktualisierung innerhalb des Beobachtungszeitraumes ein Subsegment gebildet, für das Subsegment die Features errechnet und die Pain Score abgeleitet. Der Anfangszeitpunkt jedes Subsegmentes ist der Anfang der erste Cry-Unit innerhalb des jeweiligen Beobachtungszeitraums, und das Ende des Subsegmentes das Ende letzten Cry-Unit im jeweiligen Beobachtungszeitraum. Beispielsweise erstreckt sich das bei der 3. Aktualisierung der Beobachtungszeitraum von $17.5 - 32.5$ s, das Subsegment jedoch von $19 - 32$ s aufgrund der Lage der Cry-Units. Durch die Verwendung des Beobachtungs- und Aktualisierungsintervalls wird erkennbar, dass in diesem Beispiel der Schmerzgrad innerhalb des Segmentes nach hinten hin abnimmt.

3 Visualisierung

Ziel dieses Kapitels ist es, das erarbeitete Konzept zur Visualisierung der Pain Scores vorzustellen. Das Konzept wird anhand eines Beispielsignals erläutert, welches Aufnahmen des Weinenes eines Babys enthält. Das Signal wird in Abbildung 3.1 oben gezeigt. Der Signalausschnitt ist insgesamt 220 Sekunden lang. Die Abschnitte, die Stimme des Babys enthalten, werden Schwarz dargestellt, das Hintergrundrauschen grau. Das Signal wurde segmentiert mit $t_s = 10$ s und so zwei Segmente gefunden. Die beiden Segmente sind 30.5 Sekunden voneinander entfernt.

Die Schmerzdiagnostik wird auf Basis zweier fiktiver Pain Scales durchgeführt, welche in Tabelle 3.1 definiert werden. Die „Length-Scale“ bewertet den Schmerzgrad nach der Länge des Weinenes und vergibt einen maximalen Score von 2, die „Min-Scale“ bewertet die Qualität des Weinenes und vergibt einen maximalen Score von 3.

Tabelle 3.1: Fiktive Pain Scales zur Erläuterung der Visualisierung

	„Length-Scale“	„Min-Scale“
0 Punkte	kein Weinen	Lachen
1 Punkt	kurzes Weinen	leichtes Weinen
2 Punkte	langes Weinen	mittleres Weinen
3 Punkte	-	starkes weinen

Mit Hilfe einer der in Kapitel 2 vorgestellten Strategien wurden Vorschriften zur Ableitung des Schmerz Scores auf Basis der Segmentattribute definiert. Gleichung 3.2 definiert die Funktion PS_{Length} , welche den Pain Score für Segmente nach der „Length-Scale“ ableitet. Aus der Gleichung geht hervor, dass ein Score von 0 für ein Segment nicht abgeleitet werden kann, da die Anwesenheit eines Segmentes der Abwesenheit von Weinen implizit widerspricht. Gleichung 3.2 definiert die Funktion PS_{Max} zur Schmerzableitung nach der „Min-Scale“.

$$PS_{Length}(cs) = \begin{cases} 1 & , \text{ wenn } S\text{-Length}(cs) \leq 1 \text{ min} \\ 2 & , \text{ wenn } S\text{-Length}(cs) > 1 \text{ min} \end{cases} \quad (3.1)$$

$$PS_{Min}(cs) = \begin{cases} 0 & , \text{ wenn } min_{cu}(cs) < 0.3 \text{ s} \\ 1 & , \text{ wenn } 0.3 \text{ s} \leq min_{cu}(cs) \leq 1 \text{ s} \\ 2 & , \text{ wenn } 1 \text{ s} < min_{cu}(cs) \leq 2 \text{ s} \\ 3 & \text{ sonst} \end{cases} \quad (3.2)$$

Mit Hilfe dieser Funktionen werden die Pain Scores für die beiden Segmente des Beispielsignals in Abbildung 3.1 abgeleitet. Es wurden dabei keine Aktualisierungsintervalle oder Beobachtungszeiträume genutzt. Das erste Segment nach der *Length-Scale* einen Score von

1 und nach der *Min-Scale* einen Score von 2. Das zweite Segment hat nach der *Length-Scale* einen Score von 2 und nach der *Min-Scale* einen Score von 1.

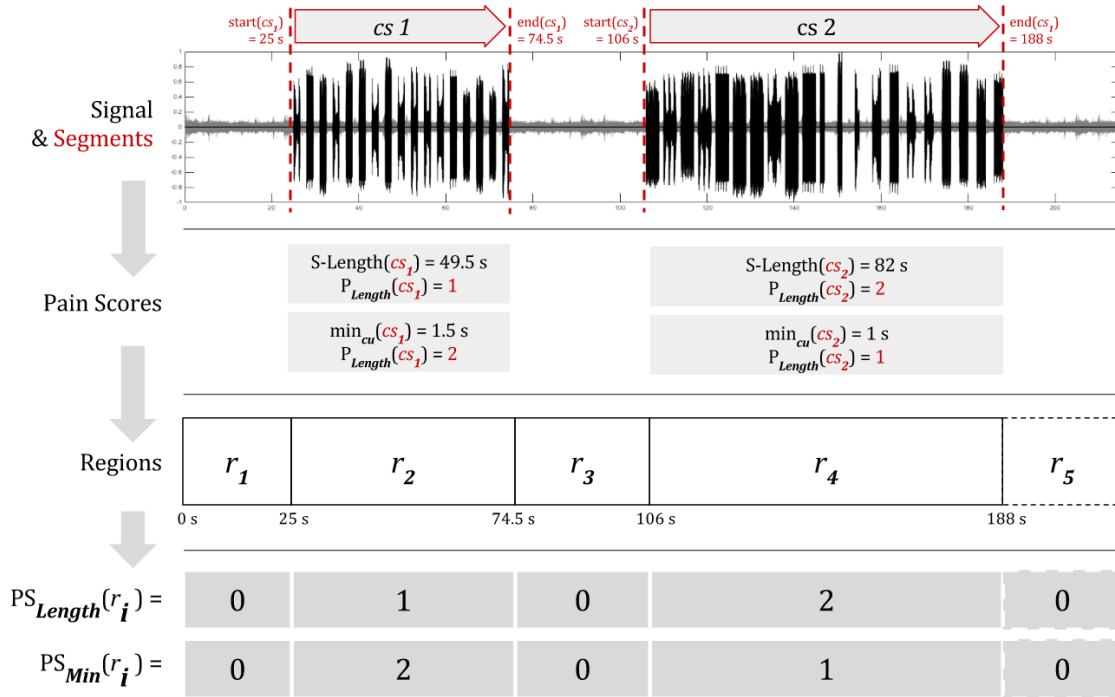


Abbildung 3.1: Oben: Ein Beispielsignal mit zwei Segmenten, die mit $t_s = 10$ s gefunden wurden. Darunter: Pain Scores, die für die beiden Segmente berechnet wurden. Darunter: Schematische Darstellung des Signals als Regionen. Unten: Scores der Regionen.

Die grundlegende Idee der Visualisierung ist, den zeitlichen Verlauf des Signals schematisch als einen *Balken* darzustellen. Dieser Balken wird in *Regionen* eingeteilt. Eine Region r beinhaltet entweder ein Cry-Segment oder den Stillebereich zwischen zwei Cry-Segmenten. Die Länge einer Region entspricht der zeitlichen Länge des jeweiligen Stille- oder Cry-Segments. In Abbildung 3.1 gibt es fünf Regionen r_1, \dots, r_5 , wobei die Region r_2 und r_4 Cry-Segmente enthalten. Die letzte Region wurde noch nicht abgeschlossen, da angenommen wird, dass das Signal weiterhin kontinuierlich eingelesen wird.

Jeder Region wird ein Pain Score zugewiesen. Enthält die Region ein Segment, so wird der Score des Segmentes für die Region übernommen. Regionen ohne Segment bekommen, unabhängig von der verwendeten Pain Scale, einen Score von 0, wie Gleichung 3.3 definiert. Es wird somit angenommen, dass ein Score von 0 bei jeder Pain Scale den Zustand „kein Schmerz“ codiert, was zumindest bei allen in Kapitel ?? vorgestellten Pain Scales der Fall ist. Abbildung 3.1 unten visualisiert diese Zuweisung von Scores zu den Regionen.

$$PS_{Scale}(r) = \begin{cases} 0 & , \text{wenn } r \text{ kein Cry-Segment beinhaltet} \\ PS_{Scale}(cs) & , \text{wenn } r \text{ ein Cry-Segment } cs \text{ beinhaltet} \end{cases} \quad (3.3)$$

Das Ziel ist es nun, jede Region mit einer Farbe einzufärben, die den entsprechenden Score anzeigt. Dazu wird für jede Pain Scale eine Funktion $F_{Scale} : S_{Scale} \mapsto P_{Scale}$ benötigt, welche einen Pain Score auf eine Farbe abbildet. Eine Abbildungsfunktion F_{Scale} wird in

diesem Zusammenhang als *Farbschema* bezeichnet und der Funktionsbereich P_{Scale} als *Farbpalette*. Ein Farbschema soll die folgenden Kriterien erfüllen:

Injektivität Jeder Score einer Scale soll anhand seiner Farbe eindeutig erkennbar sein. Dementsprechend soll gelten: $|S_{Scale}| \leq |P_{Scale}|$.

Intuitive Farbsemantik durch Ampelschema Ein Farbschema soll eine intuitive Zuordnung zwischen der Höhe des Score und der jeweiligen Farbe ermöglichen. In dieser Arbeit wurde sich für ein Ampelschema entschieden. Das heißt, dass für eine Pain Scale der jeweils niedrigste Score „grün“, der höchste Score „rot“ und ein „mittlerer“ Score als „leicht rötliches gelb“ dargestellt wird. Definiert eine Pain Scale einen maximalen Score von 2, so wie beispielsweise das FLACC-System, so ergibt sich die Abbildung $F_{FLACC}(0) = grn$, $F_{FLACC}(1) = gelb$, $F_{FLACC}(2) = rot$. Definiert die Pain Scale mehr Scores, wie beispielsweise das MBPS mit insgesamt fünf möglichen Scores, so müssen geeignete Zwischenfarben definiert werden. Daraus folgt, dass zwei Pain Scales, deren Menge an Scores gleich groß ist, das selbe Farbschema verwenden.

Visuelle Gleichabständigkeit der Farben Pain Scales definieren die Scores zwar in einer Reihenfolge, gewährleisten aber keine Vergleichbarkeit. Daher sollen die Farben eines Farbschemas eine „visuelle Gleichabständigkeit“ gewährleisten. Damit ist gemeint, dass die Farben, auf die jeweils zwei aufeinander folgende Scores einer Scale abgebildet werden, visuelle den gleichen Abstand zueinander haben. So wird verhindert, dass ein Farbschema eine Nähe oder einen Abstand zwischen Scores suggeriert, der durch die jeweilige Pain Scale nicht codiert wird.

Visuelle Gleichwichtigkeit der Farben Es kann nicht davon ausgegangen werden, dass ein bestimmter Score für die medizinische Fachkraft von größerem Interesse ist als ein anderer Score. Daher soll keine Farbe einer Farbpalette eine besondere Wichtigkeit suggerieren. Dies wird umgesetzt, in dem alle Farben einer Palette mit einer ähnlichen Buntheit definiert werden.[1]

Das Kriterium der Gleichabständigkeit legt die Verwendung des *CIELAB*-Raum zur Farbdefinition für die Farbpaletten nahe. Der Farbraum ist in Bezug auf die menschliche Farbwahrnehmung „gleichförmig“. Das heißt, dass die euklidische Distanz zwischen zwei Farben im Farbraum ihrer wahrgenommenen Unterschiedlichkeit entsprechen. Da die Farbdefinition im *CIELAB*-Raum jedoch auf für den Menschen unintuitiven Parametern beruht, wird weiterhin der *LCH*-Raum verwendet, der zyklischen Transformation des *CIELAB*-Raumes. Dieser erlaubt die Farbdefinition auf Basis der für den Menschen intuitiveren Parameter **L**uminance (Luminanz), **C**hroma (Buntheit) und **H**ue (Farbton). Dies erleichtert die Erfüllung der Farbsemantik, da sich die Farben Grün, Gelb und Rot in der H-Dimension des LCH-Raums in direkter Nachbarschaft befinden.[7][5]

Zur Zusammenstellung der konkreten Farbpaletten wird das Unterstützungswerkzeug „Lch and Lab colour and gradient picker“ von David Johnstone verwendet.¹ Das Tool erlaubt die Wahl von n Farben LCH-Raum, welche im folgenden als „Fixpunkte“ bezeichnet werden. Diese werden ihrer Reihenfolge nach als Eckpunkte eines Pfades durch den Farbraum definiert. Auf Basis dieses Pfades generiert das Tool eine Farbpalette mit m Farben. Ist $n = m$, so entspricht die Farbpalette den definierten Fixpunkten. Ist $m > n$, so findet das Tool die Zwischenfarben durch lineare Interpolation auf den Pfadkanten.

Ein „reines Rot“, codiert im RGB-Raum mit $[255, 0, 0]$, hat im LCH-Raum die Koordinaten

¹Online unter: <http://davidjohnstone.net/pages/lch-lab-colour-gradient-picker>

[53, 105, 40]. „Reines Grün“ hat die Koordinaten $RGB = [0, 255, 0]$, was $LCH = [88, 120, 136]$ entspricht. Das „leicht rötliche Gelb“ wird definiert mit $RGB = [255, 240, 0] \triangleq LCH = [93, 93, 99]$. Diese Farben haben im LCH Raum eine unterschiedliche Buntheit, weshalb sie in einer Farbpalette eine unterschiedliche Wichtigkeit suggerieren würden.[1] Daher wurde die Buntheit aller drei Fixpunkte auf den niedrigsten der drei Werte, 93, gesetzt. Die tatsächlichen Parameter der drei Fixpunkte *Rot*, *Gelb* und *Grün* sind Tabelle 3.2 zu entnehmen.

Tabelle 3.2: Fixpunkte als Basis der Farbpaletten

	L	C	H
Rot	53	93	40
Gelb	93	93	99
Grün	88	93	136

Abbildung 3.2 zeigt die Farbpaletten, die mit Hilfe des Tools für zwei bis sieben Farben auf Basis dieser Fixpunkte erstellt wurden. Bei der Farbpalette mit ungerader Farbzahl befinden sich die Fixpunkte an der ersten, letzten und mittleren Position, während die zusätzlichen Farben durch Interpolation erzeugt wurden. Bei den Farbpaletten mit gerader Farbzahl wird der mittlere Fixpunkt, das Gelb, vom Tool entfernt.

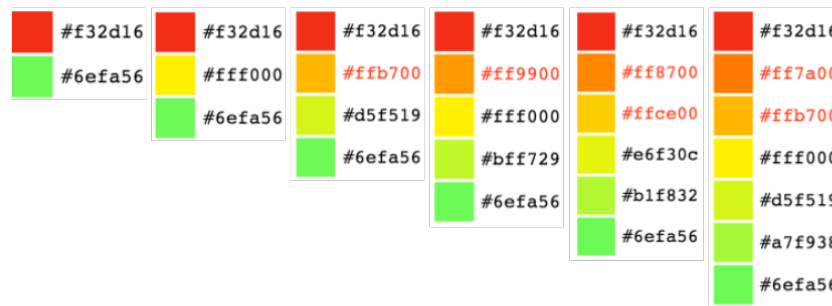


Abbildung 3.2: Erstellte Farbpaletten mit zwei bis sieben Farben inklusive der zugehörigen RGB-Koordinaten im Hex-Code.

Auf Basis der Farbpaletten wurden die Farbschemen $L_n : S_{Scale} \mapsto P_{Scale}$ definiert, abgebildet in Tabelle 3.3. Es gilt $n = |S_{Scale}|$. Die Wahl des Farbschemas zur Visualisierung einer Pain Scale richtet sich also allein nach der jeweiligen Anzahl definierter Pain Scores, also $F_{Scale} = L_{|S_{Scale}|}$. Soll also beispielsweise die „Length-Scale“ visualisiert werden, so wird das Farbschema $F_{Length-Scale} = L_3$ verwendet. Ein Pain Score von $s = 0$ wird durch das Grün mit den RGB-Koordinaten $\#6efa56$ codiert, 1 durch $\#6efa56$ und 2 durch $\#6efa56$.

Es wurden keine Farbschemen für $n < 1$ definiert, da eine Pain Scale mit nur einer Score nicht sinnvoll ist, sowie keine Farbpaletten mit $n > 7$, da keine der in Kapitel ?? gelisteten Pain Scales mehr als 7 Scores verwendet. Die Farbpaletten der Schemen S_2, S_3, S_5 und S_7 wurden direkt aus den Farbpaletten mit der jeweiligen Anzahl an Farben übernommen, die mit Hilfe des Tools von David Johnstone erzeugt wurden (Siehe Abbildung 3.2). Die durch das Tool erzeugte Farbpalette mit vier und sechs Farben wurde nicht für die Schemen S_3 und S_6 übernommen, da in ihr der Gelbe Fixpunkt nicht enthalten war. Stattdessen wurden die jeweils nächst höheren Farbpaletten unter Auslassung des ersten Grüns verwendet.

Abbildung 3.3 zeigt die Visualisierung der Schmerzdiagnostik für das Beispielsignal mit

Tabelle 3.3: Definition der Farbschemen zur Visualisierung der Pain Scales. Die Farbwerte werden als Hexadezimal-Codes für den RGB-Farbraum angegeben.

$s =$	0	1	2	3	4	5	6
$L_2(s) =$	#6efa56	#f32d16					
$L_3(s) =$	#6efa56	#ff0000	#f32d16				
$L_4(s) =$	#bffa729	#ff0000	#ff9900	#f32d16			
$L_5(s) =$	#6efa56	#bffa729	#ff0000	#ff9900	#f32d16		
$L_6(s) =$	#a7f938	#d5f519	#ff0000	#ffb700	#ff7a00	#f32d16	
$L_7(s) =$	#6efa56	#a7f938	#d5f519	#ff0000	#ffb700	#ff7a00	#f32d16

der „Length-Scale“ und der „Min-Scale“, die durch das Einfärben der in Abbildung 3.1 erläuterten Regionen entsteht. Wie bereits erläutert, verwendet die „Length-Scale“ das Farbschema L_3 . Die „Min-Scale“ definiert insgesamt 4 Scores und verwendet somit das Farbschema L_4 .

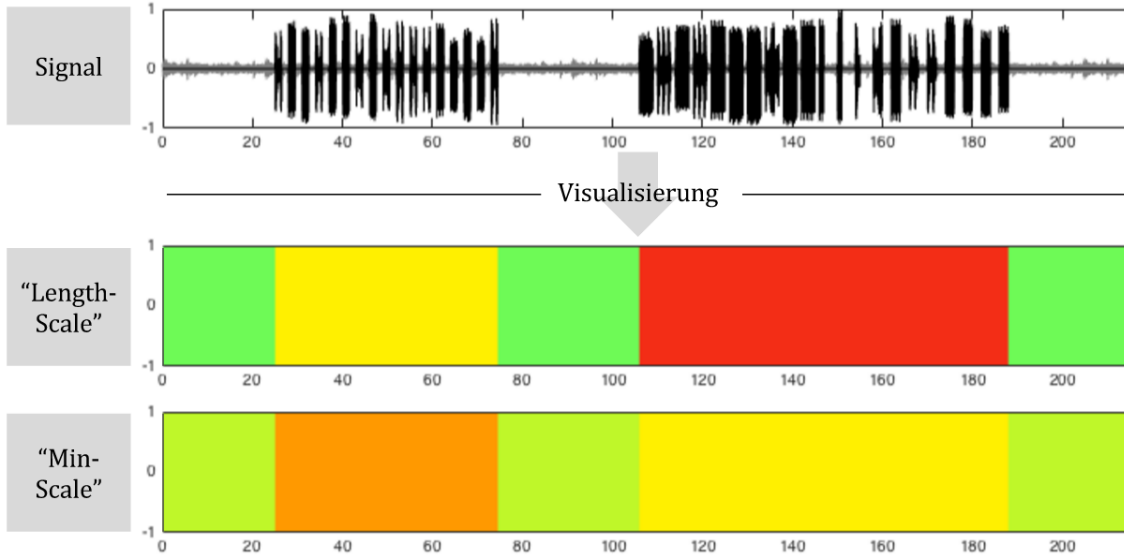


Abbildung 3.3: Visualisierung der „Length-Scale“ und der „Min-Scale“ für das Beispielsignal.

Da in diesem Beispiel kein Aktualisierungsintervall eingesetzt wurde, gibt es keine überlappenden Regionen. Abbildung 3.4 veranschaulicht, wie durch die Verwendung eines Aktualisierungsintervalls überlappende Regionen entstehen. Oben in der Abbildung ist das Beispielsignal zu sehen, welches mit $t_s = 10$ s segmentiert wurde. Es wurde ein Aktualisierungsintervall von $t_{act} = 20$ s und $t_{obs} = \infty$ verwendet. Die Aktualisierungszeitpunkte sind durch kleine Pfeile angegeben. Unten sind die aus den Subsegmenten entstandenen, einander überlappenden Regionen abgebildet. Für jede Region wird der Score angegeben, der sich nach der Diagnostik der Min-Scale ergibt, sowie jede Region nach dem Schema L_4 eingefärbt.

Die Darstellung aus Abbildung 3.4 dient nur der Veranschaulichung des Prinzips der überlappenden Regionen, eignet sich aber nicht für eine tatsächliche Visualisierung. Es werden zwei Möglichkeiten zur Visualisierung einander überlappenden Regionen vorgeschlagen:

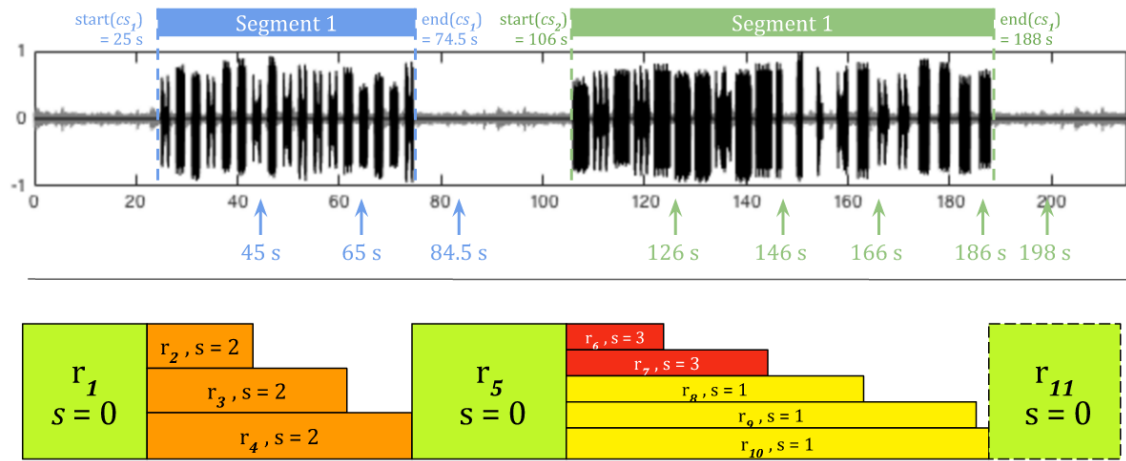


Abbildung 3.4: Oben: Beispielsignal mit zwei Segmenten. Unten: Überlappende Regionen, eingefärbt nach dem Farbschema der Min-Scale.

- Die Region mit dem späteren Endzeitpunkt wird „auf“ die Regionen mit einem früheren Endzeitpunkt gelegt. So wird der aktuell diagnostizierte Score in den Vordergrund gestellt, der zeitliche Verlauf der abgeleiteten Schmerz Scores geht jedoch verloren. Abbildung 3.5 zeigt dieses Prinzip an einem Beispiel. In diesem Fall wurde das Beispielsignal erst bis zu Sekunde 146 eingelesen, das zweite Segment ist noch geöffnet. Bei der Aktualisierung zum Zeitpunkt $t = 146\text{ s}$ wird nach der Min-Scale ein Score von 3 abgeleitet und die Region dementsprechend rot eingefärbt. Das Signal wird nun weiter eingelesen. Bei der nächsten Aktualisierung bei $t = 166\text{ s}$ wird ein Score von 1 abgeleitet und die so gelb gefärbte Region „auf“ die rot gefärbte Region gelegt.
- Die Region mit dem späteren Endzeitpunkt wird „unter“ die Regionen mit einem früheren Endzeitpunkt gelegt. So wird der zeitliche Verlauf des diagnostizierten Schmerzes in den Vordergrund gestellt. Abbildung 3.6 zeigt die Visualisierung der Schmerzdiagnostik mit der Min-Scale nach diesem Prinzip. Wie zu sehen ist, bleibt auch nach Aktualisierung der Schmerzscores des zweiten Segmentes auf 1 erkennbar, dass bei der früheren Aktualisierung ein Score von 3 abgeleitet wurde.

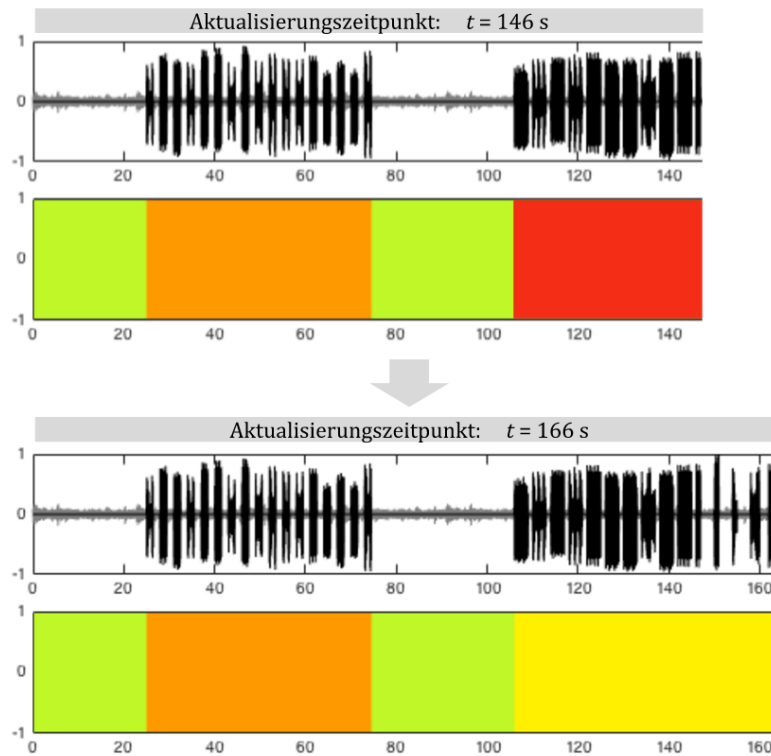


Abbildung 3.5: Visualisierung der Schmerzdiagnostik mit der Min-Scale bei Aktualisierungen. Die aktuellere Region wird *über* ältere Regionen gelegt.

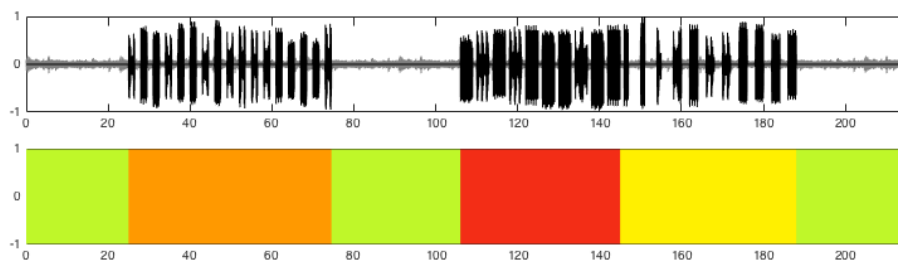


Abbildung 3.6: Visualisierung der Schmerzdiagnostik mit der Min-Scale bei Aktualisierungen. Die aktuellere Region wird *unter* ältere Regionen gelegt.

4 Zusammenfassung

Literaturverzeichnis

- [1] Alex Bigman. 6 principles of visual hierarchy for designers, 2014. Online erhältlich unter: <https://en.99designs.de/blog/tips/6-principles-of-visual-hierarchy/>.
- [2] Rami Cohen and Yizhar Lavner. Infant Cry Analysis and Detection. In *27th Convention of Electrical and Electronics Engineers in Israel*. IEEE, 2012. Online unter https://www.researchgate.net/publication/26116332_Infant_cry_analysis_and_detection.
- [3] Barbara Fuller. Acoustic Discrimination of three Cry Types. *Nursing Research*, 40(3), 1991. Online erhältlich unter: https://www.researchgate.net/publication/21125005_Acoustic_Discrimination_of_Three_Types_of_Infant_Cries.
- [4] Bonnie Stevens Huda Huijer Abu-Saad, Gerrie Bours and Jan Hamers. Assessment of pain in Neonates. *Seminars in Perinatology*, 2(5):402–416, 1998. Online unter <https://www.ncbi.nlm.nih.gov/pubmed/9820565>.
- [5] David Johnstone. Lch and lab colour and gradient picker, 2014. Online erhältlich unter: <https://en.99designs.de/blog/tips/6-principles-of-visual-hierarchy/>.
- [6] A. Rebecca Neal Linda L. LaGasse and Barry M. Lester. Assessment of infant cry: Acoustic cry analysis and parental perception. *Mental retardation and developmental disabilities*, 11(1):83–93, 2005. Online unter <https://www.ncbi.nlm.nih.gov/pubmed/15856439>.
- [7] Jarke J van Wijk Martijn Wijffelaars, Roel Vliegen and Erik-Jan van der Linden. Generating Color Palettes using Intuitive Parameters. *Computer Graphics Forum*, 40(3):743–750, 2008. Online erhältlich unter: <http://onlinelibrary.wiley.com/doi/10.1111/j.1467-8659.2008.01203.x/abstract>.
- [8] Katarina Michelsson Ole Wasz-Hockert and John Lind. Twenty-Five Years of Scandinavian Cry Research. In *Infant Crying - Theoretical and Research Perspectives*, chapter 3, pages 59 – 82. Plenung, 1985.
- [9] J L Mathew P J Mathew. Assessment and management of pain in infants. *Postgrad Med J*, 79:438–443, 2003. Online unter <http://pmj.bmj.com/content/79/934/438.full>.
- [10] Susan Parker-Price & Ronald Barr Philip Zeskind. Rythmic organization of the Sound of Infant Cry. *Dev Psychobiol*, 26(6):321–333, 1993. Online unter <https://www.ncbi.nlm.nih.gov/pubmed/8119482>.
- [11] Gyorgy Ivan Varallyay. *Analysis of the Infant Cry with Objective Methods*. PhD thesis, Budapest University of Technology and Economics, 2009. Online erhältlich unter: <https://pdfs.semanticscholar.org/5c38/b368dc71d67cbfab3077a50536b086d8eec.pdf>.

Appendices

A Auswertungsdetails VAD

Tabelle A.1: Klassifikationsgenauigkeiten der durch den REPTree entworfenen Modelle

$SNR_{Training}$	3 dB				50 dB				50+3 dB			
SNR_{Test}	3 dB	50 dB	7 dB*	Mean	3 dB	50 dB	7 dB*	Mean	3 dB	50 dB	7 dB*	Mean
Zeit	77.81%	79.02%	86.04%	80,96%	49.33%	94.70%	48.66%	64,23%	77.54%	92.47%	84.38%	84,80%
Spektrum	82.05%	89.28%	82.71%	84,68%	70.52%	94.37%	55.06%	73,31%	81.75%	91.22%	74.90%	82,62%
Cepstrum	88.98%	94.72%	92.96%	92,22%	86.83%	94.68%	92.83%	91,45%	88.98%	94.72%	92.96%	92,22%
Autokorr.	80.45%	73.47%	84.89%	79,60%	73.07%	87.14%	77.98%	79,39%	77.90%	84.88%	82.84%	81,87%
Zeit+Spektrum	82.05%	89.28%	82.71%	84,68%	70.52%	94.37%	55.06%	73,31%	81.75%	91.22%	74.90%	82,62%
Zeit+Cepstrum	88.98%	94.72%	92.96%	92,22%	86.83%	94.68%	92.83%	91,45%	88.98%	94.72%	92.96%	92,22%
Zeit+Autokorr	80.45%	73.47%	84.89%	79,60%	49.33%	94.70%	48.66%	64,23%	80.32%	92.35%	88.22%	86,96%
Spektrum+Ceps	88.98%	94.72%	92.96%	92,22%	70.65%	94.75%	55.06%	73,49%	88.98%	94.72%	92.96%	92,22%
Spektrum+Autokorr.	82.05%	89.28%	82.71%	84,68%	70.52%	95.60%	95.60%	87,24%	81.75%	94.42%	74.90%	83,69%

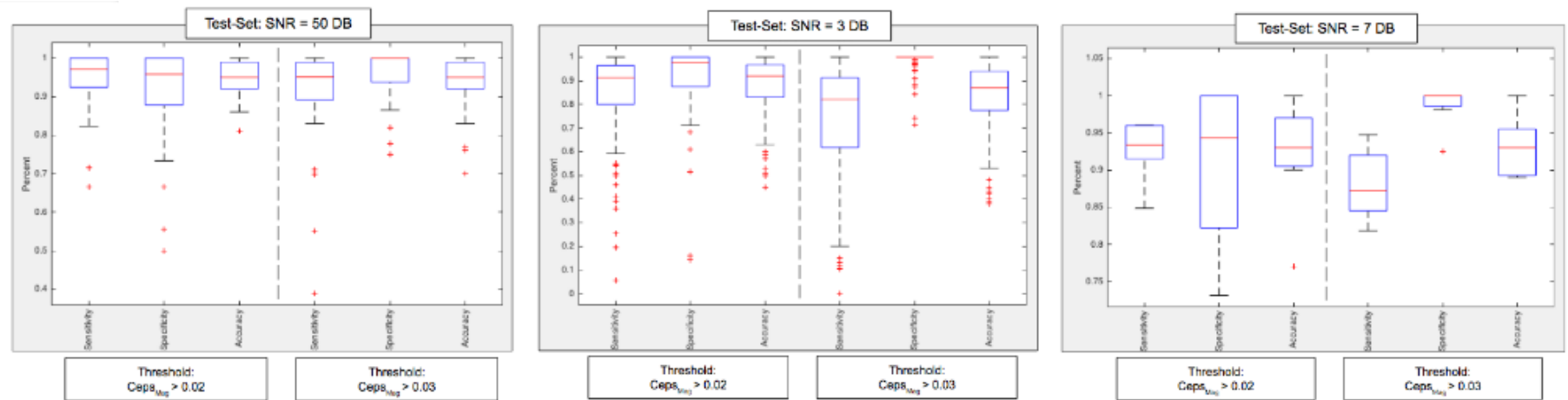


Abbildung A.1: Boxplot-Auswertung über Sensitivität, Spezifität und Genauigkeit der beiden besten Modelle