
Aus dem Institut für Neuro- und Bioinformatik
der Universität zu Lübeck
Direktor: Prof. Dr. Thomas Martinetz

# Nachweis weicher Augenfolgebewegungen auf dynamischen natürlichen Szenen

**Detection of smooth pursuit eye movements on dynamic natural scenes**

**Bachelorarbeit**
im Rahmen des Studiums Computational Life Science

Vorgelegt von
**Judith Berger**

Ausgegeben von
**PD Dr.-Ing. Erhardt Barth**
Institut für Neuro- und Bioinformatik

Betreut von
**Dipl.-Inf. Michael Dorr**
Institut für Neuro- und Bioinformatik

27. Oktober 2008

**Erklärung**

Ich versichere hiermit, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung von anderer als der angegebenen Literatur oder sonstige Hilfsmittel angefertigt habe.

Lübeck, den 24. Oktober 2008

# Inhaltsverzeichnis

# 1 Einleitung

Der Mensch bewegt seine Augen zwei bis drei mal pro Sekunde, um seine Umgebung wahrzunehmen. Dabei erweckt die Frage, worauf die Augen fokussieren und welche Bewegungen dabei erfolgen, besonderes Interesse.

Diese Frage wird durch die Klassifikation von Augenbewegungen in der aktuellen Forschung aufgegriffen. Ein Grund für dieses Interesse liegt im weiten Anwendungsbereich, den die Erkenntnisse dieses Forschungsgebietes bieten.

Um nur ein Beispiel zu nennen, könnten im Rahmen eines Versuches Testpersonen mit bestimmten Szenarien beispielsweise aus dem Straßenverkehr konfrontiert werden. Aus den gemessenen Augenbewegungen können anschließend Erkenntnisse über ihre Reaktionen auf die gegebene Situation gewonnen werden.

Mittels eines sog. Eyetrackers können die Augenbewegungen der Testpersonen aufgezeichnet werden, so dass dann festgestellt werden kann, ob sie die Besonderheiten der entsprechenden Situation - wie etwa einen rollenden Ball, das am Straßenrand spielende Kind - auch tatsächlich visuell wahrgenommen haben.

Gegenstand dieser Arbeit ist die Analyse von Augenbewegungen, die auf Grund von dynamischen Reizen entstanden sind. Für die Klassifikation der unterschiedlichen Augenbewegungen beschränkt man sich lediglich auf folgende drei Typen:

- Der erste Typ von Augenbewegung ist die sog. Sakkade.
  Hierbei handelt es sich um eine sehr schnelle Augenbewegung, die dazu dient das Auge auf eine für den Betrachter interessante Stelle auszurichten. Eine Verarbeitung von visueller Information durch das Gehirn findet zu diesem Zeitpunkt nicht bzw. nur in sehr eingeschränktem Maße statt.

- Der zweite Typ ist die sog. Fixation.
  Bei der Fixation ruht das Auge bereits auf einer interessanten Stelle. Zu diesem Zeitpunkt wird visuelle Information aufgenommen und im Gehirn verarbeitet.

- Der dritte Typ der Augenbewegungen ist die sog. weiche Augenfolgebewegung.
  Ihr Nachweis ist Gegenstand der vorliegenden Bachelor-Arbeit. Bei der weichen

> Augenfolgebewegung verfolgt das Auge ein sich bewegendes Objekt mit dem Zweck dieses, an seiner Position auf der Netzhaut, möglichst ruhig zu halten.

Fixationen und Sakkaden entstehen sowohl durch statische Reize (Bilder oder Text), als auch durch dynamische Reize (z.B. Bewegung und vorliegend Filme, die natürliche Szenen zeigen). Im Gegensatz dazu entsteht die weiche Augenfolgebewegung nur bei dynamischen Reizen.

Für die Klassifikation von Fixationen und Sakkaden existieren bereits zuverlässige Algorithmen. Im Gegensatz dazu fehlt es für die weiche Augenfolgebewegung noch an einem zuverlässig laufenden Algorithmus.
Es ist möglich Fixationen und Sakkaden in erster Näherung durch ihre Geschwindigkeit zu trennen. Dies geht bei der weichen Augenfolgebewegung nicht, da sie recht langsam ist, so dass eine Detektion durch das Rauschen in der Messung erschwert wird. Hilfsweise kann man die Detektion manuell durchführen und feststellen, ob das Auge gerade ein sich bewegendes Objekt betrachtet. Diese Lösung ist jedoch offensichtlich wenig elegant, da sie zum Einen von dem subjektiven Eindruck der Person abhängt, die die Detektion durchführt und zum Anderen extrem aufwendig ist.

Um diesen Vorgang zu automatisieren, wird in der vorliegenden Arbeit ein Algorithmus zur Bewegungsschätzung in Filmen vorgestellt, der diese Problematik lösen soll. Ferner soll eine Verknüpfung zwischen den Bewegungen der Augen, ihrer Geschwindigkeit und den Geschwindigkeiten, mit denen sich Objekte in einem beliebigen Film bewegen, hergestellt werden.

Langfristig kann dadurch bestimmt werden, ob die auf ein Objekt fokussierten Augen und das sich bewegende Objekt selbst, während des Abspielens des Filmes synchron, d.h. übereinstimmend in Punkto der Geschwindigkeit und der Richtung der Bewegung, sind.

Da bereits bekannt ist, dass das menschliche Auge weiche Augenfolgebewegungen macht, konzentriert sich diese Arbeit darauf, grob zu quantifizieren wie häufig weiche Augenfolgebewegungen auf typischen natürlichen Filmen sind.

Dazu wird im Folgenden kurz auf das menschliche Auge, dessen groben Aufbau und

dessen Bewegungen eingegangen. Ferner wird die Arbeitsweise eines Eyetrackers erläutert, konkretisiert welche Daten dieser liefert, um im Anschluss daran die Klassifikationsalgorithmen vorzustellen, durch die die Fixationen und Sakkaden klassifiziert werden können.
Im darauf folgenden Teil werden die hier verwendeten Daten und der Algorithmus zur Bewegungsschätzung vorgestellt.
Der letzte Teil dieser Arbeit beschäftigt sich mit der Auswertung der gewonnenen Daten.
Für die Auswertung wurden, neben eigenständig implementierten Programmen, zum Einen das *data source framework* und zum Anderen das *gazecom framework* genutzt. Diese wurden am Institut für Neuro- und Bioinformatik der Universität zu Lübeck implementiert.

# 2 Das Auge

In diesem Kapitel soll ein kleiner Einblick in den Aufbau des menschlichen Auges gegeben werden. Des Weiteren werden die ersten zwei der in der Einleitung genannten Augenbewegungstypen vorgestellt. Die Beschreibung der dritten Art erfolgt erst später.

## 2.1 Aufbau des Auges

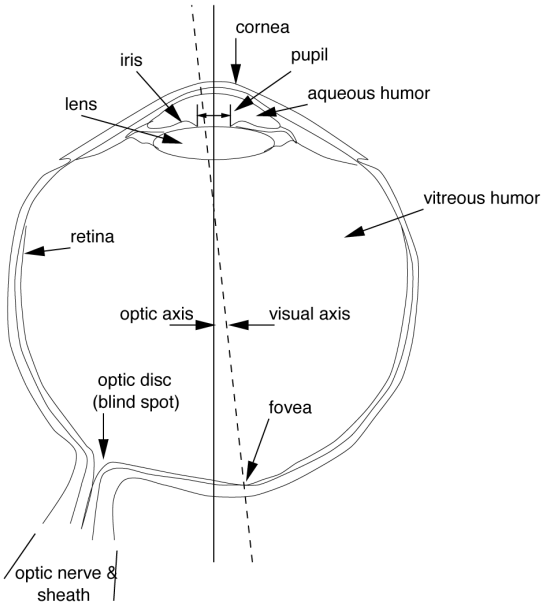


Abbildung 1: Querschnitt des menschlichen Auges. Abbildung aus [Duc00].

Die Retina bzw. Netzhaut ist eine Schicht aus spezialisiertem Nervengewebe und umgibt den Glaskörper (engl. vitreous humor), der im Innersten des Auges liegt (siehe Abbildung 1). Sie besteht aus lichtempfindlichen Rezeptoren. Diese nehmen das ins Auge fallende Licht auf und wandeln es in Nervenimpulse um. Man unterscheidet hierbei zwischen zwei Gruppen von Rezeptoren, nämlich den sog. Stäbchen und Zapfen. Die Stäbchen sind sehr empfindlich für schwaches oder dämmeriges Licht und ermöglichen uns dadurch das Sehen bei Nacht bzw. in der Dämmerung.

Die Zapfen hingegen reagieren gut auf sehr helles Licht und ermöglichen uns Dinge scharf zu sehen. Des Weiteren lassen sich Zapfen in drei Gruppen unterteilen: S-

Zapfen (short wavelength receptor), M-Zapfen (medium wavelength receptor) und L-Zapfen(long wavelength receptor). Diese besitzen unterschiedliche Maxima in der spektralen Empfindlichkeit. Daraus folgt, dass jede dieser drei Gruppen auf eine andere Farbe besonders reagiert, wodurch es dem Menschen möglich ist, Farben wahrzunehmen.

Die Fovea ist ein bestimmter Bereich der Retina, in dem die Sehschärfe des Auges am Größten ist. Sie wird auch als das „Zentrum des scharfen Sehens“ bezeichnet. Auf der Fovea werden Objekte abgebildet, die wir fixieren. Sie liegt im Zentrum der Retina, wo die Zapfendichte am höchsten ist (siehe [Geg06]). In der Peripherie nimmt die Anzahl der Zapfen immer mehr ab und die der Stäbchen immer mehr zu. Dies bewirkt, dass wir aus dem Augenwinkel, d.h. an der Peripherie des Sichtfeldes, Bewegungen wahrnehmen können, diese aber - wegen der dort befindlichen Stäbchen - nicht scharf sehen (s.o.).

## 2.2 Fixationen und Sakkaden

Bei der Analyse von Augenbewegungen, die durch statische Reize entstanden sind, kommt es lediglich auf die Fixation und die Sakkade als Typen der unterschiedlichen Augenbewegungen an.

Bei einer Fixation befindet sich das Auge in relativer Ruhe zu einem Sichtobjekt. In dieser Ruhephase, die meist $200 - 400\,\text{ms}$ lang ist, werden Informationen vom Auge aufgenommen, an das Gehirn weitergeleitet und dann dort verarbeitet. Je nachdem wie schnell eine Information verarbeitet werden kann und wieviel Information an der beobachteten Stelle vorhanden ist, variiert die Dauer einer Fixation.

Eine Sakkade hingegen ist eine sehr schnelle Bewegung, die schnellste, die der menschliche Körper ausführen kann. Sie ist ein Sprung zwischen zwei Fixationen und dient dazu, die Fovea neu zu positionieren. Die Dauer einer Sakkade liegt zwischen $20\,\text{ms}$ und $80\,\text{ms}$. Während dessen wird die Wahrnehmung visueller Information durch das Gehirn unterdrückt (sakkadische Repression), so dass wir für den Bruchteil einer Sekunde kein Bild sehen können. Dieser Mechanismus ist deshalb so wichtig, damit wir flimmerfrei ein stabiles Bild unserer Umwelt sehen können.

# 3 Eyetracking

Bevor man Augenbewegungen in die gerade beschriebenen Typen klassifizieren kann, benötigt man zunächst Daten, die die Augenbewegungen repräsentieren. Die Gewinnung dieser Daten erfolgt mittels eines Eyetrackers.

Der Eyetracker berechnet die Position der Augen bezüglich eines vorher definierten Bereiches, zum Beispiel auf einem Computermonitor. Diese Daten werden in einer bestimmten Frequenz, der sog. Abtastrate, an einen Computer geliefert. Je nachdem, wie hoch die Abtastrate ist, erhält der Computer eine bestimmte Anzahl von Gazesamples (Abtastpunkten) pro Sekunde. Die Gazesamples werden anschließend in eine Datei geschrieben, in der nun die Position der Augen gespeichert ist.

Der Eyetracker könnte beispielsweise folgendermaßen verwendet werden: Nach dem Anschluss an einen Computer wird dieser so positioniert, dass die Augen der Testperson für den Eyetracker vollständig erfassbar sind. Der Testperson können nun statische und / oder dynamische Reize angeboten werden. Im Rahmen der Betrachtung der Reize können der Testperson zusätzlich Aufgaben gestellt werden, wie beispielsweise, neben der eigentlichen Betrachtung von Bildern, auch auf bestimmte Details zu achten.
Während des Versuchsablaufs berechnet der Eyetracker fortlaufend die Position der Augen auf dem Bildschirm. Bei einer Abtastrate von zum Beispiel 250 Hz liefert dieser alle 4 Millisekunden einen x- und einen y-Wert, aus der sich die genaue Position der Augen zu diesem Zeitpunkt ermitteln lässt.

Zu jedem x,y-Wertepaar liefert der Eyetracker ferner einen Zeitstempel (den sog. time stamp). Dadurch wird ermöglicht, dass unterschiedliche Positionen in einen zeitlichen Zusammenhang gestellt werden. Außerdem erhält man noch einen “confidence value”. Dieser ist aus dem Intervall $[0, 1]$ und gibt an, ob der Eyetracker das Auge wirklich getrackt hat. Ist dies der Fall, so ist der confidence value Eins. Der Eyetracker kann die Position der Augen nun sicher angeben und das Gazesample wird durch die Eins als gültig markiert.

Ist der confidence value Null, dann hat der Eyetracker das Auge nicht getrackt. Ursa-

chen hierfür können z.B. das Blinzeln der Testperson oder das Rauschen des Trackers sein. Dies wiederum hat zur Folge, dass nicht davon ausgegangen werden kann, dass die vom Eyetracker gelieferten x- und y-Werte richtig sind. Daher kennzeichnet die Null das Gazesample als ungültig.

## 3.1 Rohdaten

Diese vier Werte (time stamp, x, y und confidence value) werden zeilenweise in eine Datei geschrieben. Diese Datei ist ein sog. Gazecoord file und ist immer vom gleichen Format:
Die ersten Zeilen bilden den sog. Header, in dem Informationen zum Versuchsaufbau festgehalten sind. Dazu zählen der Abstand der Testperson zum Bildschirm und die Höhe und Breite des vordefinierten Bereichs (in Metern), bezüglich dessen der Eyetracker die Position der Augen bestimmen soll. Im vorliegenden Fall also die Größe des Bildschirms. Des Weiteren sind die Auflösung des Bildschirms (in Pixeln) und Kommentare enthalten. Diese Daten sind deshalb wichtig, weil die Geschwindigkeit des Gazes an Stelle von Pixel pro Sekunde in Grad pro Sekunde angegeben werden soll. Nur dann ist die Gazegeschwindigkeit nämlich distanzunabhängig (siehe Abschnitt 3.2).

Nach dem Header folgen in vier Spalten aufgeteilt die oben genannten Werte. Die Anzahl der Zeilen in einem Gazecoord file hängt davon ab, wie lange der Eyetracker die Bewegungen aufzeichnet.

## 3.2 Umrechnung der Geschwindigkeit

Für die Berechnung der Geschwindigkeit mit der die Augen Objekte wahrnehmen (Gazegeschwindigkeit), kommt es darauf an, welche Strecke das Auge in einer bestimmten Zeit zurücklegt. Die Größe Zeit wird hierbei meist in Sekunden angegeben. Die Größe Strecke bzw. Weg entspricht der Distanz zweier Samplingpunkte, die vom Eyetracker geliefert werden. Diese haben x,y-Koordinaten. Da jedoch die Betrachtungsfläche ein Monitor ist, entsprechen diese dem Pixel $(x, y)$ des Monitors.

Die Einheit der Gazegeschwindigkeit wäre demnach Pixel pro Sekunde. Es ist jedoch möglich die Geschwindigkeit noch in einer anderen Einheit anzugeben: Grad pro Sekunde.

Die Angabe der Gazegeschwindigkeit in Pixel pro Sekunde ist im Gegensatz zu ihrer Angabe in Grad pro Sekunde nachteilig, weil sie stets in Verbindung mit der Distanz der Testperson vom Bildschirm anzugeben ist.

Dies entfällt jedoch bei der Angabe der Geschwindigkeit in Grad pro Sekunde, weil die Distanz durch die Umrechnung in das Ergebnis mit einbezogen wird, so dass die Angabe der Geschwindigkeit distanzunabhängig ist.

Diese Umrechnung lässt sich anhand der Abbildung 2 näher verdeutlichen:

Gesucht ist ein Faktor, mit dem die berechneten Geschwindigkeiten des Gazes multipliziert werden können, so dass das Ergebnis statt in Pixel pro Sekunde in Grad pro Sekunde ist.

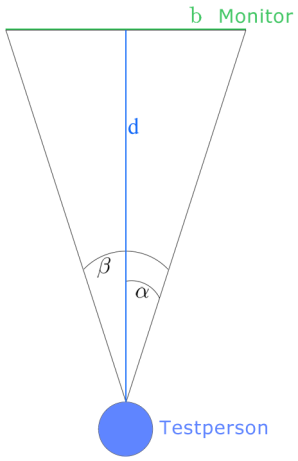


Abbildung 2: Skizze zum Versuchsaufbau beim Eyetracking, Sicht von oben.

Durch den Header sind die Breite des Monitors $b$ und die Distanz zwischen Testperson und Monitor $d$ gegeben. Als erstes wird der Winkel $\alpha$ bestimmt, der in dem rechtwinkligen Dreieck mit den Katheten der Länge $d$ und $\frac{b}{2}$ liegt.

Aus der Trigonometrie ist bekannt: $\tan(\alpha) = \frac{b}{2d}$

$\Rightarrow \qquad \alpha = \arctan\left(\frac{b}{2d}\right).$

Des Weiteren ist bekannt, dass $\beta$ der Winkel ist, der das gesamte Blickfeld einschließt und dass für $\beta$ gilt: $\beta = 2\alpha$

$\Rightarrow \qquad \beta = 2\arctan\left(\frac{b}{2d}\right).$

Aus dem Header ist die Auflösung des Bildschirms, d.h. seine Breite in Pixeln, bekannt. Diese sei jetzt $w$. Ferner gilt: $\beta$ [Grad] $\widehat{=}$ $w$ [Pixel]. Daraus kann der Faktor für die Umrechnung hergeleitet werden:

$$\begin{aligned} & \beta \text{ [Grad]} && \widehat{=} && w \text{ [Pixel]} \\ \Leftrightarrow\ & \tfrac{\beta}{w} \text{ [Grad]} && \widehat{=} && 1 \text{ [Pixel]}. \end{aligned}$$

Diese Gleichung kann nun mit jedem beliebigen Wert multipliziert werden. Somit kann auch jede Geschwindigkeit in Pixel pro beliebige Zeiteinheit eingesetzt werden. Als Ergebnis erhält man den entsprechenden Wert in Grad pro beliebige Zeiteinheit.

# 4 Algorithmen zur Analyse der Rohdaten (statischer Reiz)

Wie im vorangegangenen Kapitel bereits dargestellt, geben die Daten des Eyetrackers Aufschluss darüber, wo sich die Augen zu einem bestimmten Zeitpunkt $t$ befinden. Daher kann man die Daten als Funktion $f(t) = \vec{x}$, mit $\vec{x} = (x, y)^{\mathrm{T}}$, darstellen, die die x- und y-Position der Augen gegen die Zeit zeigt. In Abbildung 3 ist, wegen der einfacheren Darstellbarkeit, lediglich die x-Position des Auges gegen die Zeit aufgetragen.

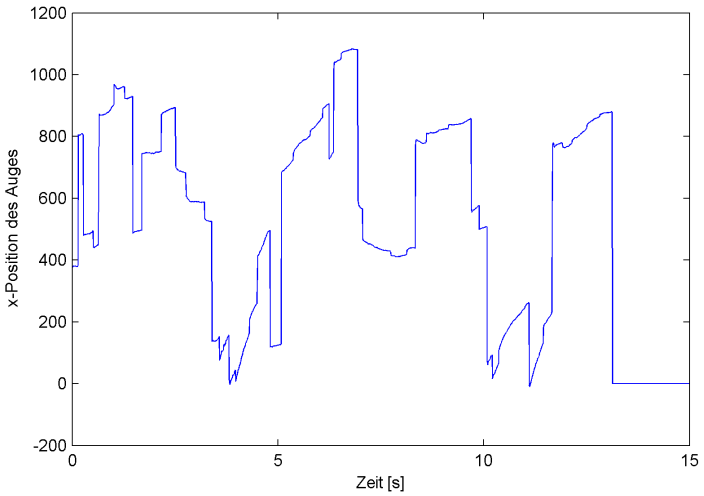


Abbildung 3: x-Position des Auges aufgetragen gegen die Zeit. Die vertikalen Bereiche sind Sakkaden, die horizontalen Fixationen.

Fixationen und Sakkaden unterscheiden sich in dieser Betrachtung besonders durch ein Merkmal: die Geschwindigkeit des Auges.
Betrachtet man die Position des Auges über die Zeit, wie in Abbildung 3, betrachtet man implizit die Geschwindigkeit des Auges. Bei der oben gezeigten Darstellung stellt man fest, dass sich die Position bei einer Fixation fast nicht verändert (das Signal ist in etwa stationär), wobei dies im Gegensatz dazu bei einer Sakkade sehr stark der Fall ist (das Signal steigt steil an bzw. fällt ab). Findet man in einem Signal, wie dem oben gezeigten, die Stellen, an denen bezüglich der Position eine starke Veränderung zu erkennen ist, so kann man Fixationen von Sakkaden unter-

scheiden.

Eine andere Möglichkeit ist die Geschwindigkeit des Gazes explizit zu berechnen, um Fixationen und Sakkaden zu unterscheiden: Bei einer Fixation sind die Bewegungen, die das Auge macht, klein, d.h. die Geschwindigkeit ist sehr gering. Bei einer Sakkade hingegen ist die Bewegung sehr groß und damit die Geschwindigkeit hoch. Findet man in dem nun betrachteten Signal die Stellen, an denen sich die Geschwindigkeit stark verändert, kann man ebenfalls Fixationen und Sakkaden voneinander trennen.

Aus diesem Ansatz ergeben sich folgende zwei Algorithmen: die Dwell-Time Methode und die Velocity-Detection Methode.

## 4.1 Dwell-Time Methode

Bei der Dwell-Time Methode wird ein Zeitfenster über die Daten geschoben, dessen Länge einen akzeptablen Wert für die Dauer einer Fixation angibt. In dem Zeitfenster wird der Mittelwert $\mu$ des Signals berechnet. Für den Algorithmus wird eine Entfernung $D$ zu $\mu$ vorgegeben. $N$ sei die Anzahl aller Punkte, die in dem Zeitfenster liegen (theoretisch somit alle zu einer Fixation gehörenden Punkte).

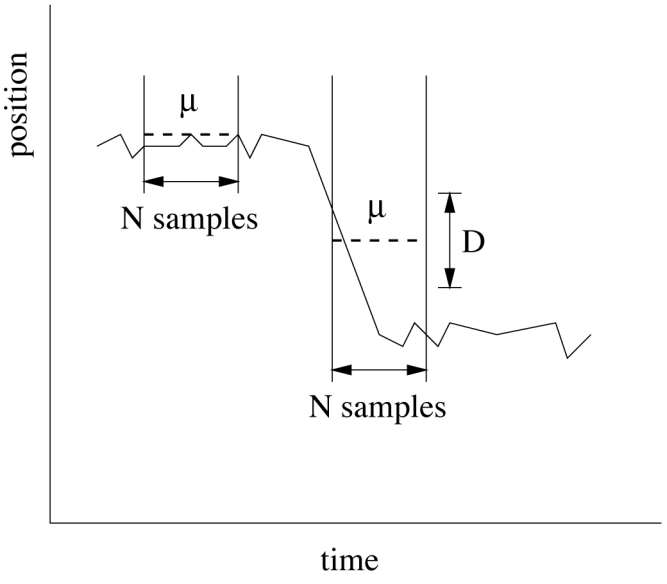


Abbildung 4: Illustration der Dwell-Time Methode, Abbildung aus [Duc00].

Falls

- $M$ von $N$ Punkten innerhalb der Entfernung $D$ zu $\mu$ liegen, so wird in diesem Fenster eine Fixation klassifiziert (siehe linker Teil von Abbildung 4),
- nicht $M$ von $N$ Punkten innerhalb der Entfernung $D$ zu $\mu$ liegen, so wird in diesem Fenster eine Sakkade klassifiziert (siehe rechter Teil von Abbildung 4).

Die Werte für $M$, $N$ und $D$ bestimmt man empirisch.

## 4.2 Velocity-Detection Methode

Bei der Velocity-Detection Methode wird innerhalb eines Zeitfensters der Betrag der Geschwindigkeit des Signals berechnet ($v$). Zwischen zwei jeweils aufeinander folgenden Samples berechnet sich $v$ nach physikalischen Grundlagen durch die Veränderung des Ortes, vorliegend nach dem euklidischen Abstand der beiden Punkte, geteilt durch die Veränderung der Zeit: $v = \frac{\sqrt{(x_{t+1} - x_t)^2 + (y_{t+1} - y_t)^2}}{\Delta t}$.
Anschließend vergleicht man $v$ mit einem vorgegebenen Schwellwert $T_{crit}$.
Falls

- $v \leq T_{crit}$ wird eine Fixation klassifiziert,
- $v > T_{crit}$ wird eine Sakkade klassifiziert.

Sowohl die Dwell-Time, als auch die Velocity-Detection Methode ist nachzulesen in [Duc00].

Zusätzlich zu diesen Methoden werden sog. Plausibilitätsprüfungen durchgeführt. Dadurch wird überprüft, ob beispielsweise die Dauer einer Bewegung oder auch die berechnete Geschwindigkeit sinnvoll ist. Ist dies nicht der Fall, so werden die betroffenen Gazesamples weder einer Sakkade noch einer Fixation zugeordnet. Es gibt damit auch Abschnitte in den Daten, die weder als Fixation noch als Sakkade klassifiziert werden können. Ein Grund dafür ist, dass das menschliche Auge neben diesen zwei auch weitere Bewegungen macht. Dies liegt daran, dass beispielsweise angenommen werden könnte, dass sich das Auge während einer Fixation nicht bewegt. Dies ist jedoch nicht der Fall, da das Auge in Wirklichkeit kleinste Bewegungen

macht und sich somit nicht in einem Zustand vollkommener Ruhe befindet. Diese kleinsten Bewegungen werden als Tremor bezeichnet. Man erkennt diese minimalen Bewegungen am Verlauf des oben gezeigten Signals (siehe Abbildung 3):
Dieser ist nicht parallel zur x-Achse (Fixation), steigt senkrecht an bzw. ab (Sakkade), um die nächste stationäre Phase (Fixation) zu erreichen. Der Verlauf zeigt viele minimale An- und Abstiege.

Als weiterer Grund für die Nichtzuordnung eines Bereiches der Signals zu einer Fixation bzw. Sakkade kommt in betracht, dass die Daten durch das Rauschen des Eyetrackers verfälscht werden.

Durch die Velocity-Detection Methode und die Dwell-Time Methode liegen zwei Algorithmen vor, die aus den Eyetrackerdaten die Positionen von Sakkaden und Fixationen bestimmen können. Diese beiden Algorithmen dienen jedoch nicht nur der Klassifikation von Sakkaden und Fixationen im Fall eines statischen Reizes, sondern auch der bei dynamischen Reizen. In Augenbewegungen, die auf dynamischen Reizen entstanden sind, gibt es zusätzlich zu den o.g. Augenbewegungstypen die weiche Augenfolgebewegung. Im folgenden Kapitel geht es darum heraus zu finden, wie weiche Augenfolgebewegung in den Rohdaten (Gazecoord files) zu finden ist.

# 5 Analyse der Rohdaten (dynamischer Reiz)

Grundlage für die weiche Augenfolgebewegung ist zunächst, dass sich eine Testperson einen Film ansieht, in dem sich Personen oder Gegenstände bewegen. In diesem Moment kann die Testperson ein sich bewegendes Objekt fixieren und dieses mit ihren Augen verfolgen. Durch diese Folgebewegung verbleibt das Objekt in etwa an der gleichen Position auf der Netzhaut. Die dabei entstehende Bewegung des Auges ist von gleichmäßiger Geschwindigkeit. In Folge der visuellen Verfolgung des Objekts kommt es nicht zum Wechselspiel zwischen Fixationen und Sakkaden.

Der Versuch die gleiche Bewegung ohne ein sich bewegendes Ziel nachzuahmen, würde daran scheitern, dass das menschliche Auge eine entsprechende weiche und gleichmäßige Bewegung nicht ohne ein sich bewegendes Ziel durchführen kann. Dies hätte zur Folge, dass diese Bewegungen als Abfolge von Fixationen und Sakkaden zu klassifizieren wären.

Selbst wenn in dieser Arbeit die Augenfolgebewegung zunächst nur nachgewiesen und noch nicht direkt klassifiziert wird, so sind die im folgenden Abschnitt erklärten Verfahren und Algorithmen ebenso zur Klassifikation erforderlich.

Die weiche Augenfolgebewegung ist daher schwer zu klassifizieren, weil die Geschwindigkeit dabei kein so eindeutiges Merkmal darstellt, wie dies bei Sakkaden und Fixationen der Fall ist (siehe Kapitel 4). Außerdem darf man für die Klassifikation der weichen Augenfolgebewegung nicht nur die Augenbewegungen isoliert betrachten, sondern muss auch die Bewegungen im Film mit einbeziehen. Nur wenn sich tatsächlich ein Objekt im Film bewegt und die Augen auch tatsächlich auf diese Stelle blicken, besteht die Möglichkeit, dass eine Augenfolgebewegung entsteht.

Für den Nachweis einer Augenfolgebewegung muss zum Einen die Geschwindigkeit des Gazes berechnet werden. Diese Berechnung ist unkompliziert, da der Eyetracker die genauen Positionen der Augen liefert (siehe Kapitel 3). Zum Anderen muss die Bewegung in den Filmen detektiert werden. Dies erfolgt mittels Algorithmen zur Bewegungsschätzung. Die beiden vorliegend vorgestellten Algorithmen schätzen für jedes Pixel in jedem Frame des Videos die Bewegung. Aus diesen Informationen

werden neue Filme generiert, in denen binär kodiert ist an welchen Stellen Bewegung geschätzt wurde. Die Bewegungsschätzung und das Generieren der neuen Filme bilden den komplexeren Teil des Nachweises.

## 5.1 Die Algorithmen

Bevor der ST-Algorithmus (structure tensor) und der MST-Algorithmus (minors of the structure tensor) zur Bewegungsschätzung in Filmen erläutert werden, werden zunächst die theoretischen Grundlagen der Algorithmen näher erklärt:

Die Filme seien gegeben durch die Funktion $f(x, y, t)$. Das heißt, für jedes Pixel $(x, y)$ zu jedem Zeitpunkt $t$ liefert die Funktion $f$ einen Wert (bei Farbbildern eine vektorielle Größe, bestehend aus drei Komponenten: Y (Lichtstärke), U und V (Farbanteile); bei Graustufen eine Intensitätskomponente).

$$\mathbf{D}(x, y, t) = (f_x, f_y, f_t)^{\mathrm{T}} (f_x, f_y, f_t) = \begin{pmatrix} {f_x}^2 & f_x f_y & f_x f_t \\ f_y f_x & {f_y}^2 & f_y f_t \\ f_t f_x & f_t f_y & {f_t}^2 \end{pmatrix}$$

ist die Matrix, die aus dem Produkt des Gradienten von $f$ mit sich selbst hervorgeht.

Durch Faltung von $\mathbf{D}$ mit einem gauß'schen Weichzeichner $h(x, y)$, wird in einer 2D-Umgebung (mit $h(x, y, t)$ in einer 3D-Umgebung) weichgezeichnet, und es entsteht

$$\mathbf{J}(x, y, t) = h(x, y) * \mathbf{D}(x, y, t). \tag{1}$$

$\mathbf{J}$ heißt Strukturtensor (ST) von $f$.

Des Weiteren sei $\mathbf{M} = Minors(\mathbf{J})$ die Matrix mit den Elementen $M_{ij}$ $(i, j = 1, 2, 3)$, wobei $M_{ij}$ die Determinante ist, die durch Streichen der $4-i$ten Zeile und der $4-j$ten Spalte von $\mathbf{J}$ entsteht.

### 5.1.1 ST-Algorithmus

Der Algorithmus schätzt mittels der Eigenwerte des Strukturtensors $\mathbf{J}(x, y, t)$ aus Gleichung ( 1 ) den zu dem Pixel $(x, y)$ gehörenden Bewegungsvektor $\vec{v}$.

Dazu werden als erstes die Eigenwerte $\lambda_i$ von $\mathbf{J}$ berechnet. Da $\mathbf{J}$ symmetrisch und positiv semidefinit ist, sind alle Eigenwerte reell und man kann sie der Größe nach sortieren: $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$. Die entsprechenden Eigenvektoren bilden eine Orthonormalbasis und spannen das prinzipielle Koordinatensystem der lokalen 3D Raum-Zeit Umgebung $U$ bezüglich der Bewegung auf. Dieses muss nicht mit dem Weltkoordinatensystem übereinstimmen. Bei dem für diese Arbeit interessanten Fall ist $\lambda_3 = 0$ und $\lambda_1, \lambda_2 > 0$. Dieser tritt genau dann auf, wenn es eine Translation in x,y-Richtung gibt, die geschätzt werden soll. In der Praxis ist $\lambda_3$ allerdings nie echt gleich Null, sondern nur annähernd. Zum Einen liegt dies am Rauschen des Eyetrackers und zum Anderen daran, dass die Geschwindigkeit der Objekte im Film nicht immer gleichmäßig ist. Beschleunigt dort ein Objekt, so sind alle drei Eigenwerte von Null verschieden.

Als nächstes wird geprüft, ob

1. der Eigenwert $\lambda_1 > T_\lambda$;

   Dadurch wird der Algorithmus gegen Rauschen robuster und es wird garantiert, dass es eine Signaländerung (Bewegung) in wenigstens einer Richtung gibt. Nach [Bar00] ist $T_\lambda = 20$. Für den im Abschnitt 5.2 verwendeten MST-Algorithmus ist dieser Wert jedoch nicht relevant.

2. das Konfidenzmaß $\quad c = \left(\frac{\lambda_1 - \lambda_3}{\lambda_1 + \lambda_3}\right)^2 - \left(\frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2}\right)^2 > T_c,$

   Aus den Eigenwerten berechnet sich das Konfidenzmaß $c \in [0, 1]$, das die Sicherheit in der Bewegungsschätzung anzeigt. Je näher das Konfidenzmaß an der Eins liegt, desto größer ist die Sicherheit, dass eine Signaländerung vorliegt. Im Gegensatz dazu liegt eher keine Signaländerung vor, wenn dieses sich der Null annähert.

   Ist $c > T_c$ so kann davon ausgegangen werden, dass eine Situation im Film vorliegt, die zumindest ähnlich dem gewünschten Idealfall von $\lambda_3 = 0$ und $\lambda_1, \lambda_2 > 0$ ist. Wenn 1. erfüllt ist, so ist $\lambda_1$ mindestens 20 $\Rightarrow$ falls $\lambda_3 = 0$ *und* $\lambda_2 = 0$, so ist $c = 0$ $\left(\text{denn } \left(\frac{20}{20}\right) - \left(\frac{20}{20}\right) = 0\right)$ und damit würde der Algorithmus nicht ausgeführt werden.

Nimmt man jedoch an, dass $\lambda_1 = \lambda_2 > T_\lambda \Rightarrow \left(\frac{\lambda_1 - 0}{\lambda_1 + 0}\right) - \left(\frac{0}{\lambda_1 + \lambda_2}\right)$. Damit ist $c = 1$, somit liegt mit absoluter Sicherheit eine Signaländerung in zwei Raumdimensionen vor ($\lambda_1, \lambda_2 > 0$), die jedoch zeitlich konstant ist ($\lambda_3 = 0$).

Je mehr sich $\lambda_2$ und $\lambda_1$ gleichen, desto kleiner wird der zweite Bruch der zuletzt genannten Gleichung und umso mehr nähert sich $c$ der Eins an. In diesem Fall steigt die Sicherheit, dass eine Signaländerung in zwei Richtungen vorliegt.

Ist 1. und 2. erfüllt, so wird der Eigenvektor $\vec{r}_3 = (r_{31}, r_{32}, r_{33})$ zum Eigenwert $\lambda_3$ berechnet. Das ist der Vektor, der in die Richtung zeigt, in der keine Änderung des Signals vorliegt. Nimmt man die x-Komponente von $\vec{r}_3$ und teilt sie durch die t-Komponente von $\vec{r}_3$, so bekommt man die Signaländerung in x-Richtung, d.h. die x-Komponente des Bewegungsvektors $\vec{v}$. Analog erhält man die y-Komponente von $\vec{v}$. Durch Faltung mit einem gauß'schen Weichzeichner $h_{\vec{v}}$ erhält man den endgültigen Bewegungsvektor

$$\vec{v}_{ST} = h_{\vec{v}} * \left(\frac{r_{31}}{r_{33}}, \frac{r_{32}}{r_{33}}\right).$$

Dieser stellt die Richtung und Stärke der Bewegung in diesem Pixel dar. Als Ergebnis der Anwendung des ST-Algrithmus erhält man für die Originalfilme für jedes Frame ein Bild ähnlich der Abbildung 5.

Abbildung 5: Ein Frame eines mit dem MST-Algorithmus analysierten Films. Jeder Pfeil zeigt die Richtung und Stärke der Bewegung in diesem Pixel an.

Der Algorithmus wird nur dann durchgeführt, wenn im aktuellen Frame die Unterdeterminante $M_{11} > T_{M_{11}}$, wobei $T_{M_{11}}$ 1% des Maximums von $M_{11}$ über alle Frames entspricht. Dadurch wird der Algorithmus weniger empfindlich für verrauschte Daten.

### 5.1.2 MST-Algorithmus

Für den MST-Algorithmus gelten die gleichen theoretischen Grundlagen wie für den ST-Algorithmus. Der MST-Algorithmus unterscheidet sich jedoch in der Berechnung des Bewegungsvektors $\vec{v}$. Er nutzt dazu nicht die Eigenwerte, sondern spezielle Beziehungen der Unterdeterminanten von $\mathbf{J}$. Man hat vier unterschiedliche Beziehungen gefunden, die jeweils einen Bewegungsvektor liefern:

$$\left(\frac{M_{31}}{M_{11}}, \frac{-M_{21}}{M_{11}}\right) = \vec{v}_1 \qquad \left(\frac{M_{33}}{M_{13}}, \frac{-M_{23}}{M_{13}}\right) = \vec{v}_3$$

$$\left(\frac{M_{23}}{M_{12}}, \frac{-M_{22}}{M_{12}}\right) = \vec{v}_2 \qquad \left(\frac{M_{33}}{M_{11}}, \frac{-M_{22}}{M_{11}}\right) = \left(v_{4x}^2, v_{4y}^2\right).$$

Die Herleitung dieser Beziehungen ist nachzulesen in [BF00]. Es sei jedoch erklärend dazu gesagt:
Wenn ein Film $f(x, y, t)$ betrachtet wird, in dem sich ein Muster mit konstanter Geschwindigkeit $\vec{v} = (v_x, v_y)$ bewegt, dann erfüllt $f$ folgende Bedingung:
$f(x, y, t) = f(x - v_x t, y - v_y t)$. Weil die Geschwindigkeit konstant über die Zeit ist, liegt nun lediglich eine Signaländerung in x,y-Richtung vor, d.h. $f$ ist lokal nur noch zweidimensional. Falls $f$ die eben genannte Bedingung erfüllt, dann gilt $\vec{v} = \vec{v}_1 = \vec{v}_2 = \vec{v}_3 = \vec{v}_4$. In der Praxis sind geringe Abweichungen zulässig, indem man $\vec{v} \approx \vec{v}_i$ fordert. Genauer gesagt lässt man eine Abweichung um einen Schwellwert $\theta$ zu: $\vec{v} = \vec{v}_i \pm \theta$.

Der Algorithmus startet die Berechnung der Bewegungsvektoren $\vec{v}_i$ entsprechend den oben genannten Gleichungen, falls der jeweilige Nenner größer als 1% des Maximums über alle Nenner dieses Frames ist. Anschließend wird überprüft, ob $v_{ix}^2 + v_{iy}^2 > T_m^2$, wobei $T_m$ 1% der maximalen Länge von $\vec{v}_1$ in dem jeweiligen Frame entspricht. So wird sichergestellt, dass die Vektoren eine bestimmte Länge haben und nicht jedes Rauschen als Bewegung gewertet wird. Abschließend werden die einzelnen $\vec{v}_i$

miteinander verglichen. Je ähnlicher sie sich in ihrer Ausrichtung sind, umso höher ist die Sicherheit, dass die geschätzte Bewegung für dieses Pixel richtig ist. Genauer gesagt wird überprüft, ob die Winkelabweichung zwischen den $\vec{v}_i$ weniger als $T_\theta$ ist. Ist dies der Fall, dann berechnet sich der Bewegungsvektor als Mittel der $\vec{v}_i$ gefaltet mit einem gauß'schen Weichzeichner $h_v$:

$$\vec{v}_{MST} = h_v\,(x, y) * \frac{(\vec{v}_1 + \vec{v}_2 + \vec{v}_3 + \vec{v}_4)}{4}.$$

Ist die Winkelabweichung größer als $T_\theta$, so wird $\vec{v}_{MST}$ auf Null gesetzt.
Der Wert für $T_\theta$ war 15 Grad.

Als Ergebnis der Anwendung des MST-Algorithmus erhält man ein für jedes Frame ein Bild wie in Abbildung 5 gezeigt.

Diese Bilder sind für die Klassifikation von weichen Augenfolgebewegungen sehr wichtig, da die Bewegungen der Augen sich mit den Pfeilen vergleichen lassen. Damit kann verifiziert werden, ob die Bewegungsrichtung der Augen mit derjenigen der Pfeile übereinstimmt. Ist dies der Fall, dann liegt eine Augenfolgebewegung vor.
Für den Rahmen dieser Arbeit ist v.a. die Information hilfreich, *dass* Bewegung in einem Pixel geschätzt wurde.
Im folgenden Abschnitt wird ein Programm vorgestellt, welches unter Verwendung des MST-Algorithmuses für jedes Pixel angibt, ob tatsächlich Bewegung geschätzt wurde oder nicht.

## 5.2 Velocity Movies

Das in C++ implementierte Programm *velocity* erzeugt binär kodierte sog. „velocity movies". Diese haben die gleiche Größe wie die Originalfilme, d.h. die Auflösung des Bildes und die Anzahl der Frames sind gleich. *Velocity* schätzt für jedes Pixel in jedem Frame des Filmes die Geschwindigkeit mittels des MST-Algorithmuses. Übersteigt die berechnete Geschwindigkeit in einem Pixel einen angegebenen Schwellwert, so bekommt das Pixel des velocity movies den Wert 255 zugewiesen. In diesem Fall

wird das Pixel weiß dargestellt, wodurch hervorgehoben wird, dass Bewegung geschätzt wurde.
Ist die Geschwindigkeit unterhalb des Schwellwertes, so wird dem Pixel des velocity movies der Wert Null zugewiesen. Das Pixel wird schwarz dargestellt. Daraus folgt, dass in diesem keine Bewegung geschätzt wurde. Der Schwellwert wird relativ zur maximalen Geschwindigkeit im Film festgelegt.

Abbildung 6: Die linke Seite zeigt ein Frame eines Originalfilmes, die rechte Seite den zugehörigen Frame des velocity movies. Hierbei sind in weiß die Pixel mit und in schwarz die Pixel ohne geschätzte Bewegung dargestellt.

Bevor ein velocity movie erzeugt wird, wird über verschiedene Orts-Zeit-Skalen regularisiert. Das heißt, dass die Geschwindigkeit erst für jedes Pixel im Originalfilm geschätzt wird, dann zum Beispiel nur noch für jedes Pixel bei halber räumlicher Auflösung bzw. bei einem Viertel zeitlicher Auflösung. Dabei entspricht natürlich ein Pixel pro Frame bei halber räumlicher Auflösung zwei Pixeln pro Frame im Originalfilm bzw. ein Frame bei einem Viertel der zeitlichen Auflösung vier Frames bei voller zeitlicher Auflösung. Der Grund für die Betrachtung der Filme auf den unterschiedlichen Orts-Zeit-Skalen ist der, dass man durch die geringere örtliche Auflösung kleine Bewegungen herausfiltert und größere Bewegungen in Relation dazu hervorgehoben werden. Bei geringerer zeitlicher Auflösung hat dieses Vorgehen Auswirkungen auf kurzzeitige Bewegungen, diese werden dadurch abgeschwächt. Länger andauernde Bewegungen werden hervorgehoben.

Dadurch, dass die geschätzten Geschwindigkeiten in jedem Pixel auf verschiedenen Auflösungsstufen addiert werden, werden somit Stellen, an denen tatsächlich Bewe-

gung stattfindet, hervorgehoben. Die Addition der Geschwindigkeiten in den Pixeln geschieht momentan unabhängig von der Richtung der Geschwindigkeit. In diesem Fall ist dies jedoch ausreichend, da in erster Linie interessant ist, *dass* Bewegung vorhanden ist und weniger in welche Richtung sie geht.

Der eben beschriebene Ablauf des Programms bedeutet, dass der Film zweimal durchlaufen werden muss. Beim ersten Durchlauf werden die Geschwindigkeiten für jedes Pixel und die Maximalgeschwindigkeit berechnet. Beim zweiten Durchlauf werden die Geschwindigkeiten jedes Pixels auf den unterschiedlichen Auflösungsstufen addiert und den Pixeln des velocity movies dem Schwellwert entsprechend, die Werte 255 oder Null zugewiesen.

# 6 Auswertung

Das Ziel dieser Arbeit ist herauszufinden, ob in den Gazedaten Folgebewegungen enthalten sind. Dazu werden die Geschwindigkeiten des Gazes berechnet und anschließend wird verglichen, wie oft der Gaze auf einem statischen Pixel (Pixel ohne geschätzte Geschwindigkeit) bzw. auf einem dynamischen Pixel (Pixel mit geschätzter Geschwindigkeit) ruhte. Diese Informationen werden anschließend in sog. velocity files gespeichert (siehe Abschnitt 6.3). In diesem Kapitel werden erst kurz die zur Auswertung verwendeten Daten erläutert und anschließend ihre Aufbereitung und Weiterverarbeitung erklärt.

## 6.1 Erläuterung der Daten

Die folgende Auswertung wurde auf Augenbewegungsdaten gemacht, die auf 18 verschiedenen Testvideosequenzen aufgenommen wurden.

Abbildung 7: Einzelne Frames aus vier unterschiedlichen Filmen. Links oben: Menschen in der Fußgängerzone; Rechts oben: Kreisverkehr; Links unten: Tauben; Rechts unten: Schiffe.

Die Filme wurden mit einer JVC JY-HD10 HDTV Videokamera aufgenommen und zeigen unterschiedliche natürlich Szenen in und um Lübeck. (7 Filme: Menschen in der Fußgängerzone, am Strand, beim Spielen im Park; 4 Filme: viel befahrene Straßen und Kreisverkehre; 4 Filme: Tiere; 3 Filme: Filme, die beinahe einen Stilllebencharakter haben, z.B. ein vorbeifahrendes Schiff in der Ferne.) Jeder Film ist ca. 20 Sekunden lang, hat eine Auflösung von 1280 zu 720 Pixeln (entspricht dem Bildformat 16:9) und eine Bildfrequenz von 29.97 Bildern pro Sekunde (progres - sive scan). Im Allgemeinen war die Videokamera für die Aufnahmen fixiert. Einige Filme enthalten jedoch leichte Kamerabewegungen, weil dies die betreffende Szene erforderte.

Die Filme zeigen natürliche Szenen und keine synthetischen, wie sie in der Sehforschung meist verwendet werden. Der Unterschied zwischen den beiden ist v.a., dass Bewegungen in synthetischen Szenen leicht zu kontrollieren und somit auch vorherzusagen sind. Bei natürlichen Szenen ist dies nicht der Fall. Dies liegt daran, dass hier Bewegungen nicht künstlich generiert werden und somit Zeit und Ort einer Bewegung nicht automatisch bestimmbar sind.

Die Filme wurden auf einem Monitor gezeigt, dessen Fläche 40 cm mal 30.6 cm groß war. Die Filme bedeckten dabei eine Fläche von 39.8 cm mal 22.8 cm. Die Teile des Monitors, die nicht von dem Film bedeckt waren, wurden schwarz dargestellt. Der Abstand zwischen den Testpersonen und dem Bildschirm betrug 45 cm. Der Film füllte damit ein horizontales Sichtfeld von etwa 48 Grad aus. Die Testpersonen wurden zwar angewiesen die Filme aufmerksam zu betrachten, erhielten ansonsten jedoch keine weitere Aufgabe. Die Augenbewegungen wurden mit einer Abtastrate von 250 Samples pro Sekunde aufgenommen. Benutzt wurde der kommerzielle videobasierte Eyetracker Eyelink II, hergestellt von SR Research.
Für jeden Film wurden Aufnahmen von 54 Testpersonen gemacht. Über den confidence value wurden ungültige Samples markiert (siehe Abschnitt 3). Augenbewegungsaufnahmen, die mehr als 5% ungültige Samples enthielten, wurden verworfen. So blieben 37 bis 52 Aufnahmen pro Video, insgesamt 844 Gazecoord files, für die Auswertung.

## 6.2 Aufbereitung der Daten

Vor der Auswertung der Gazedaten musste zunächst das Rauschen in den Daten selbst verringert werden, weswegen die Gazecoord files räumlich und zeitlich gefiltert wurden. Dazu wurden alle Gazesamples innerhalb eines Zeitintervalls von 300 ms betrachtet, dessen Zentrum das aktuelle Gazesample mit dem zugehörigen time stamp $t$ war. Aus dieser Menge wurden alle Samples entfernt, deren Abstand zum aktuellen Sample $(x_t, y_t)$ größer war als ein vorgegebener Maximalabstand von 23 Pixeln.

**M**: Menge aller Punkte deren time stamp $ts_i \in [t \pm 150\,\text{ms}]$ und für deren Position $(x_i, y_i)$ gilt $x_i \in [x_t \pm 23\,\text{Pixel}]$ und $y_i \in [y_t \pm 23\,\text{Pixel}]$

Die Filterparameter - d.h. der zeitliche Schwellwert und der maximale Abstand - entsprechen denjenigen, die für die Bewegungsschätzung in den Filmen benutzt wurden. Die 300 ms als zeitlicher Schwellwert ergeben sich daraus, dass bei den Filmen über 9 frames à 33.367 ms, also $\approx 300$ ms, gefiltert wurde.

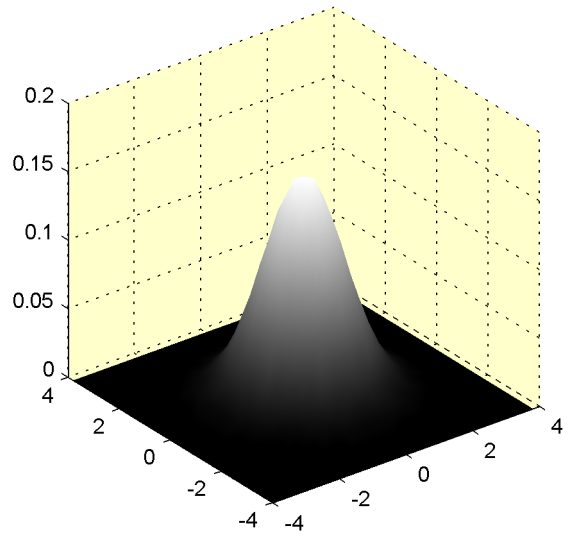


Abbildung 8: Visualisierung einer zweidimensionalen Gausverteilung

Die 23 Pixel als räumlicher Schwellwert ergeben sich als Näherung an die Werte, mit dem die Filme räumlich gefiltert wurden (visualisiert in Abbildung 8). Es wurde

zweimal mit einem räumlichen Maximalabstand von 5 Pixeln und einmal mit einem räumlichen Maximalabstand von 15 Pixeln gefiltert. Durch die Faltung ergeben sich $(5 + 5 - 1) + 15 - 1 = 23$ Pixel.

Das aktuelle Gazesample wird durch eine gewichtete Summe über die in der Menge **M** verbleibenden Gazesamples ersetzt. Die dabei verwendeten Gewichte folgen ebenfalls einer Gaußverteilung.

$$x_t = \sum x_i \alpha_i \,,\; y_t = \sum y_i \alpha_i$$

$$\text{mit } (x_i, y_i) \in \mathbf{M} \text{ und } \alpha_i = \exp\left(\left(\frac{ts_i - \mu_{ts}}{\sigma_{ts}}\right)^2 + \left(\frac{x_i - \mu_x}{\sigma_x}\right)^2 + \left(\frac{y_i - \mu_y}{\sigma_y}\right)^2\right).$$

Wobei $\mu_{ts} = t,\; \mu_x = x_t,\; \mu_y = y_t$ und $\sigma_{ts} = 47000\,\text{ms},\; \sigma_x = 3.4\,\text{Pixel},\; \sigma_y = 3.4\,\text{Pixel}$.

Des Weiteren werden mittels des Sakkadendetektors aus dem *data source framework* die Sakkaden aus den Gazedaten herausgefiltert. Dass heißt der zugehörige confidence value wird auf Null gesetzt und das Sample somit als ungültig markiert. Sakkaden zeichnen sich durch eine sehr hohe Geschwindigkeit aus und sind für das Ziel, weiche, gleichmäßige Folgebewegungen zu klassifizieren nicht relevant. Ungültige Gazesamples werden weder für die Filterung der Gazedaten, noch für spätere Berechnungen betrachtet.

Aus den Filmen werden mit dem Programm *velocity* velocity movies erzeugt. Diese sind von der selben Größe wie die Originalfilme, d.h. die Auflösung des Bildes und Länge des Filmes sind identisch (siehe Abschnitt 5.2).

## 6.3 Erzeugung der velocity files

Nachdem die Aufbereitung der Daten abgeschlossen ist, werden diese an das selbst implementierte Programm übergeben. Es erzeugt daraus sog. „velocity files“. Für die Erzeugung eines velocity files wird die Geschwindigkeit jedes einzelnen Gazesamples berechnet und der Wert des velocity movies an genau dieser Stelle ermittelt.

Dem Programm werden beim Aufruf ein velocity movie, alle Gazefiles, die auf dem Originalfilm entstanden sind, und eine Integrationszeit übergeben. Die Integrationszeit sagt aus, wie groß das Intervall ist, in dem die Geschwindigkeit der Gazesamples

berechnet werden soll. Die Geschwindigkeit, die für jedes einzelne Gazesample berechnet werden muss, wird dann nicht zwischen dem aktuellen und dem nächsten Punkt, sondern innerhalb eines Intervalls mit der angegebenen Länge berechnet. Sinn und Zweck der Angabe der Integrationszeit ist die Mittelung der Gazegeschwindigkeiten. Dadurch können kurze, schnelle Bewegungen den Übrigen derart angeglichen werden, dass über das gesamte Intervall eine gleichmäßige Geschwindigkeit besteht. Eine eindeutige Geschwindigkeitsänderung ist dann ein Hinweis darauf, dass eine gleichmäßige, weiche Augenfolgebewegung beendet ist.

Der Wert des Pixels im velocity movie wird folgendermaßen ermittelt: Man nimmt sich das aktuelle Gazesample und den dazu gehörigen Frame des velocity movies. Zugehörig bedeutet in diesem Fall, dass der time stamp des Gazesamples größer/gleich dem des Frames und echt kleiner dem des nächsten Frames ist, also
$ts_{frame_t} \leq ts_{gaze} < ts_{frame_{t+1}}$.

Nun betrachtet man das Pixel an der Stelle des Gazesamples: $frame\,(gaze_x, gaze_y)$. Der Wert des Pixels gibt an, ob an dieser Stelle eine Bewegung geschätzt wurde oder nicht. Dieser Wert wird in das velocity file geschrieben.
Im Idealfall ist der Wert des Pixels Null oder 255. Ist er Null, so haben wir keine Bewegung im Film an dieser Stelle. Ist der Wert des Pixels 255, dann liegt eine Bewegung vor. Leider unterscheidet sich die Realität vom Idealfall, da es an den Kanten zwischen schwarzen und weißen Pixeln teilweise zu MPEG-Kompressionsartefakten kommt. Das bedeutet, dass in diesem Fall dem betreffenden Pixel ein Wert zwischen Null und 255 zugewiesen wird. Deshalb werden Werte, die kleiner als 127 als „Nicht-Bewegung“ und diejenigen, die größer/gleich 127 sind als „Bewegung2 gewertet.

In dem velocity file, welches nun sukzessive entsteht, besteht jede Zeile aus drei Einträgen. An der ersten Stelle wird der time stamp notiert, dann folgt die Geschwindigkeit des Gazesamples und an dritter Stelle kommt der Wert des Pixels des velocity movies.

## 6.4 Auswertung der velocity files

Ein sich bewegendes Objekt nimmt beim Abspielen des Filmes auf dem Monitor stets eine bestimmte Anzahl von Pixeln ein, für die im Anschluss mittels des MST-Algorithmuses eine Bewegung geschätzt wird. Diese Pixel werden als dynamisch bezeichnet. Blickt eine Testperson auf ein dynamisches Pixel, lässt sich daraus schließen, dass sie im Originalfilm ein sich bewegendes Objekt betrachtet hat, es an dieser Stelle demnach zu einer weichen Augenfolgebewegung gekommen ist.

Neben den dynamischen Pixeln existieren allerdings auch statische Pixel. Diese sind diejenigen Pixel, für die mittels des MST-Algorithmuses keine Geschwindigkeit geschätzt wurde.

Für die Frage, wie hoch der Anteil der weichen Augenfolgebewegung ist, kommt es darauf an, wie oft der Blick der Testperson auf einem dynamischen oder statischen Pixel ruhte.

Um dies zu unterscheiden, werden die Gazegeschwindigkeiten in den velocity files in zwei Gruppen aufgeteilt: In der ersten Gruppe sind die Gazegeschwindigkeiten, bei denen im Pixel Bewegung geschätzt worden war: $Pixel \geq 127$.
In der zweiten Gruppe sind diejenigen, bei denen der Gaze auf einem Pixel ohne Bewegung geruht hat: $Pixel < 127$.

Mit Hilfe des Programms MATLAB erzeugt man einen Vektor, dessen Komponenten die Anzahl der Gazesamples einer bestimmten Geschwindigkeit enthalten. Dies erfolgt für die beiden Gruppen von Gazegeschwindigkeiten. Würde man diese Vektoren plotten, erhielte man ein Histogramm. Die darin enthaltenen Balken (Klassen) repräsentieren jeweils eine Gazegeschwindigkeit. Um von der Länge eines Balkens den prozentualen Anteil dieser Gazegeschwindigkeit ablesen zu können, werden der Vektoren auf die Länge eins normiert.

Anschließend subtrahiert man die beiden Vektoren voneinander: „Gaze auf dynamischem Pixel“ – „Gaze auf statischem Pixel“ und plottet diesen, durch die Subtraktion entstandenen Vektor als Histogramm (Differenzhistogramm). Der Vorteil der Subtraktion gegenüber dem direkten visuellen Vergleich der geplotteten Vektoren liegt

darin, dass durch die Subtraktion kleinste Unterschiede deutlicher hervortreten und damit tatsächlich erkannt werden können.

Falls die Klassen des Histogrammms in positive Richtung zeigen, liegen mehr Gazesamples dieser Geschwindigkeit auf dynamischen als auf statischen Pixeln. Sollten jedoch mehr Gazesamples einer Geschwindigkeit auf statischen Pixeln liegen, dann zeigt die entsprechende Klasse in negative Richtung. Falls der Fall auftreten sollte, dass die Anzahl von Gazesamples auf dynamischen und statischen Pixeln gleich ist, dann wird - wegen der o.g. Subtraktion - diese Klasse (Balken) nicht dargestellt.

In Abbildung 9 ist abschließend eine Übersicht über die durchgeführten Arbeitsschritte gegeben.

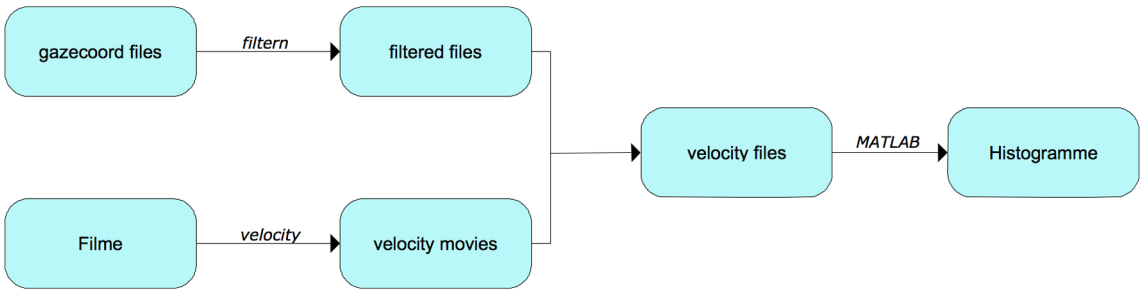


Abbildung 9: Übersichtsdiagramm zur Darstellung der einzelnen Schritte

# 7 Resultate

Für die Auswertung der Gazedaten wurden zehn unterschiedliche Integrationszeiten gewählt: $10, 20, 30, \ldots, 100$ ms. Zum Schluss wird verglichen, welche Integrationszeit für die Auswertung der Augenbewegungsdaten bzgl. der weichen Augenfolgebewegung am besten geeignet ist. Für jede der zehn Integrationszeiten sind die Daten mit der in Abschnitt 6.3 und 6.4 erläuterten Vorgehensweise analysiert worden. In der Visualisierung der Ergebnisse werden die zehn Differenzhistogramme, die zu einem Film gehören, unter- bzw. nebeneinander gezeigt (siehe dazu die Abbildungen 10, 11 und 12).

Aus der Abbildung 10 ist ersichtlich, dass jedes der Differenzhistogramme Balken aufweist, die in negative Richtung zeigen. Diese Balken ergeben sich daraus, dass mehr Gazesamples dieser Geschwindigkeit auf statischen als auf dynamischen Pixeln liegen. Daraus lässt sich schließen, dass statische Pixel tatsächlich vorhanden waren und damit ist verbunden, dass der Film statische Objekte enthielt, die von der Testperson fixiert wurden. Anderenfalls würden im Differenzhistogramm keine Balken auftauchen, die in negative Richtung zeigen. Bei näherer Betrachtung des Differenzhistogramms bzgl. dieser Balken fällt auf, dass diese in einem Geschwindigkeitsbereich zwischen Null und einem Grad pro Sekunde liegen.

Balken, die Geschwindigkeiten über einem Grad pro Sekunde repräsentieren, zeigen hingegen in positive Richtung. Dies steht dafür, dass ein größerer Anteil der Gazesamples auf dynamischen Pixeln die diesen Balken zugeordnete Geschwindigkeit haben. Daraus folgt, dass der Film dynamische Pixel enthielt. Wäre dies nicht der Fall, würden die Balken der entsprechenden Geschwindigkeiten nicht in positive Richtung zeigen. Die Testpersonen fixierten somit letztendlich dynamische Pixel. Damit entstanden auf dem Film „Kreisverkehr“, bei Geschwindigkeiten von mehr als einem Grad pro Sekunde, weiche Augenfolgebewegungen.

Bei Betrachtung der Abbildung 11 fällt auf, dass die Ausrichtung der Balken gegenüber derjenigen in Abbildung 10 in umgekehrter Richtung verläuft: bei Gazegeschwindigkeiten im Bereich von Null bis ca. 1.3 Grad pro Sekunde zeigen die Balken in positive Richtung. Bei Gazegeschwindigkeiten ab 1.3 Grad pro Sekunde zeigen sie

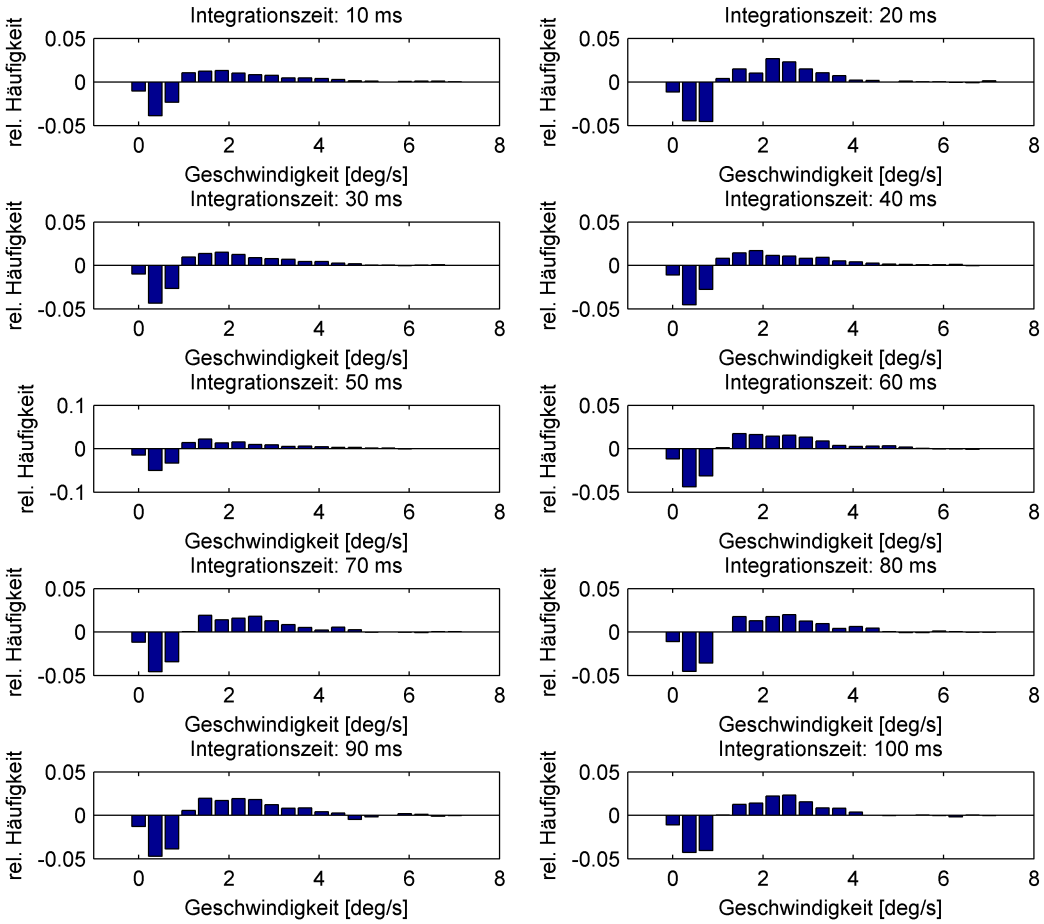


Abbildung 10: Differenzhistogramme der unterschiedlichen Integrationszeiten des Filmes „Kreisverkehr“. Der Schwellwert bei der Erzeugung des velocity movies war 1‰ der Maximalgeschwindigkeit.

hingegen in negative Richtung. Aus den vorangegangenen Schlussfolgerungen gilt für diesen Histogrammverlauf, dass Gaze mit sehr niedriger Geschwindigkeit auf dynamischen Pixeln positioniert ist, während Gaze mit größerer Geschwindigkeit auf statischen Pixeln liegt. Die weiche Augenfolgebewegung entstand somit im Bereich niedriger Gazegeschwindigkeiten. Daher wurden Objekte mit den Augen verfolgt, die sich sehr langsam bewegten.

Sowohl in Abbildung 10 als auch in Abbildung 11 ist auffällig, dass trotz der verschiedenen Integrationszeiten kaum ein Unterschied im Verlauf der jeweiligen Diffe-

renzhistogramme erkennbar ist.

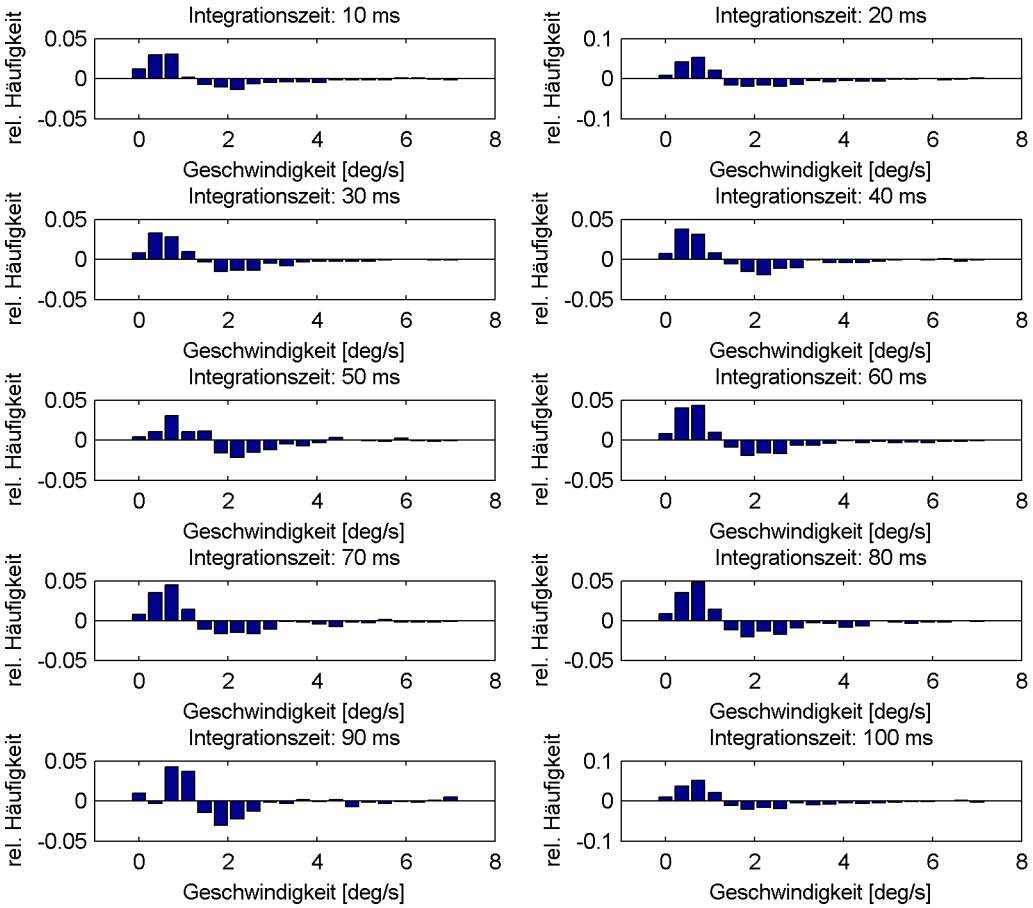


Abbildung 11: Differenzhistogramme der unterschiedlichen Integrationszeiten des Filmes „Brücke“. Der Schwellwert bei der Erzeugung des velocity movies war 1‰ der Maximalgeschwindigkeit.

Einen von den bisherigen Differezhistogrammen gänzlich abweichenden Verlauf zeigen die Differenzhistogramme in Abbildung 12. Im Gegensatz zu den vorangegangenen („Harmonischer Verlauf“), beschreiben diese Differenzhistogramme einen scheinbar willkürlichen Verlauf („Willkürlicher Verlauf“). Dies ist damit zu erklären, dass nicht automatisch erwartet werden kann, dass die Gazesamples sich in verschiedene Geschwindigkeitsbereiche einteilen lassen, in denen entweder weiche Augenfolgebewegung stattfindet oder statische Pixel betrachtet wurden. Da diese Differenzhisto-

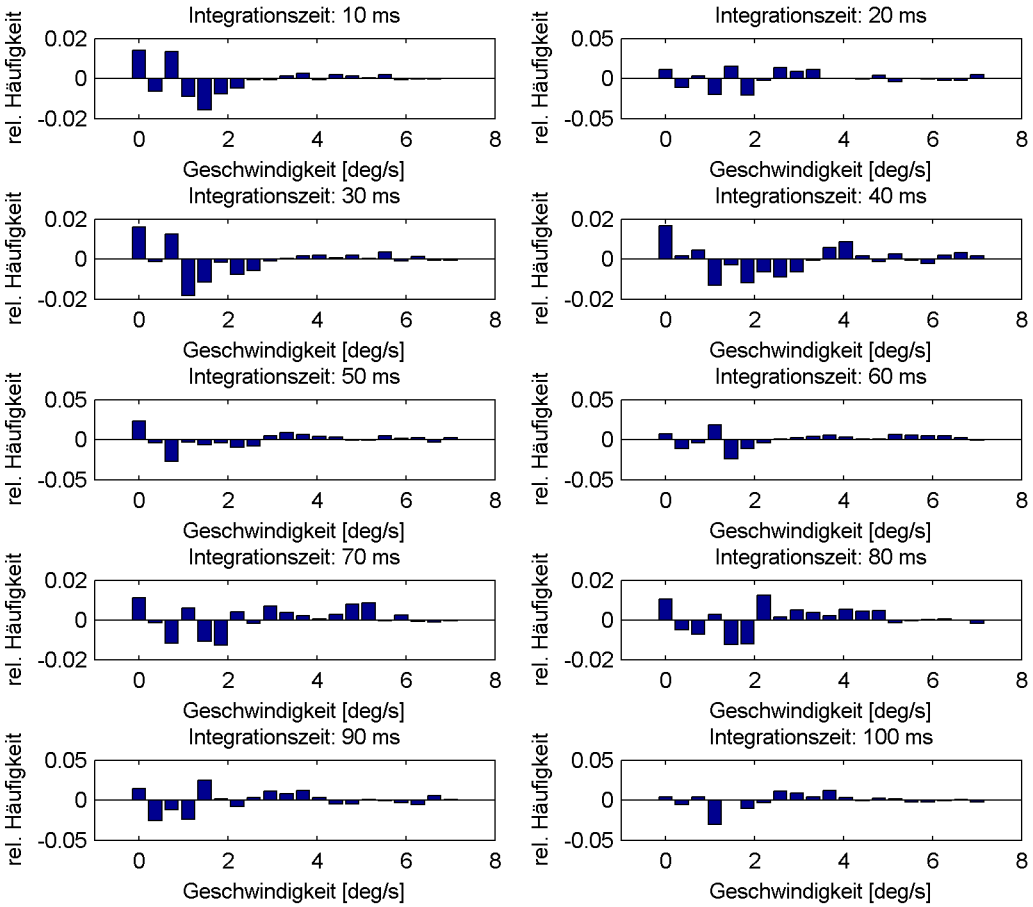


Abbildung 12: Differenzhistogramme der unterschiedlichen Integrationszeiten des Filmes „Holstentor“. Der Schwellwert bei der Erzeugung des velocity movies war 0.1‰ der Maximalgeschwindigkeit.

gramme, die einen solchen „willkürlichen Verlauf“ zeigen, aber nicht systematisch sind, ist davon auszugehen, dass der Grund für diese Fälle nur zufällig stark verrauschte Daten sind und sie nicht durch weiche Augenfolgebewegung hervorgerufen wurden.

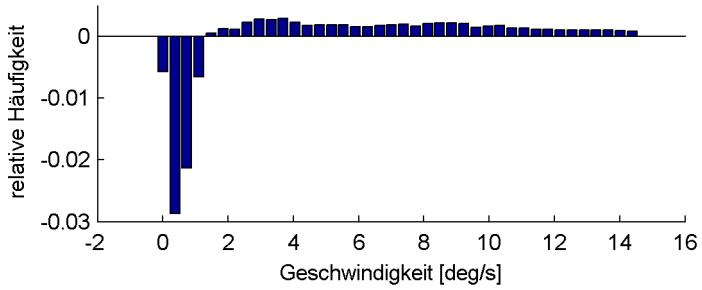


Abbildung 13: Differenzhistogramm von allen Filmen mit allen Integrationszeiten, als Einheit abgebildet.

Für das in Abbildung 13 gezeigte Differenzhistogramm wurden die Histogrammvektoren der Gazesamples auf dynamischen Pixeln und diejenigen der Gazesamples auf statischen Pixeln von allen Filmen mit allen Integrationszeiten addiert. Anschließend wurden die zwei, durch diese Addition entstandenen Vektoren auf die Länge Eins normiert und ihre Differenz geplottet. Bei diesem Differenzhistogramm sieht man, dass Gazesamples, für die langsame Geschwindigkeiten berechnet wurden (0 bis etwa 1.5 Grad proSekunde), sehr oft auf nicht bewegten Pixeln im Film positioniert sind. Gazesamples mit Geschwindigkeiten die größer als 1.5 Grad pro Sekunde, liegen meist auf dynamischen Pixeln. Die Geschwindigkeiten, mit denen weiche Augenfolgebewegung stattfand, variieren stark. Dies liegt daran, dass die Filme verschiedene Histogrammvektoren aufweisen, weil die Objekte in den jeweiligen Filmen unterschiedlich schnell sind.

# 8 Zusammenfassung

Das Ziel dieser Arbeit war es nachzuweisen, ob bei der Betrachtung dynamischer natürlicher Szenen weiche Augenfolgebewegungen auftreten. Ferner war damit die Frage verbunden, in welchem Geschwindigkeitsbereich diese liegen.

Dazu wurden Augenbewegungsdaten analysiert, die auf natürlichen Szenen aufgezeichnet wurden. Die Positionen der Augen wurden in sog. Gazecoord files gespeichert. Anschließend wurden diese gefiltert, um zum Einen das Rauschen darin zu verringern und zum Anderen, um Sakkaden als ungültig zu markieren, welche für die Klassifikation von weicher Augenfolgebewegung nicht relevant sind. Aus den Filmen wurden mittels des Programms *velocity* binär kodierte sog. velocity movies erzeugt. In ihnen wird festgehalten, an welchen Stellen des Filmes Bewegung geschätzt wurde. Dazu wird der MST-Algorithmus genutzt, der für jedes Pixels in jedem Frame die Bewegung schätzt.
Die gefilterten Gazecoord files wurden jeweils mit dem zugehörigen velocity movie an das selbständig implementierte Programm übergeben, welches daraus sog. velocity files erzeugt hat. In den velocity files werden zu jedem Gazesample folgende Informationen festgehalten: Der zugehörige time stamp, die Geschwindigkeit des Gazesamples und ob dieses Gazesample auf einem statischen oder dynamischen Pixel ruhte.

Die Daten aus den velocity files wurden anschließend in zwei Gruppen aufgeteilt. Der ersten Gruppe wurden die Gazegeschwindigkeiten der Gazesamples zugeordnet, die auf einem dynamischen Pixel lagen. Der zweiten Gruppe wurden dementsprechend diejenigen Gazegeschwindigkeiten der Gazesamples zugewiesen,die auf statischen Pixeln lagen. Aus jeder Gruppe wurde jeweils ein Histogrammvektor erzeugt. Die Komponenten dieser Vektoren repräsentieren die unterschiedlichen Gazegeschwindigkeiten.

Im Anschluß wurde die Differenz der beiden Histogrammvektoren gebildet und geplottet. Dadurch wird ermöglicht, dass einfacher verglichen werden kann, ob in der jeweiligen Geschwindigkeitsklasse mehr Gazesamples auf dynamischen oder auf statischen Pixeln lagen.

Lagen bei der Auswertung der Histogramme mehr Gazesamples einer bestimmten Gazegeschwindigkeit auf dynamischen Pixeln, dann lag an dieser Stelle der Gazegeschwindigkeit eine weiche Augenfolgebewegung vor.

Abschließend kann gesagt werden, dass diese Arbeit die Grundlagen für die Klassifikation der weichen Augenfolgebewegung erweitert. Sofern dieses Thema in Zukunft näher untersucht werden sollte, reicht es dann nicht mehr aus - wie im vorliegenden Fall - zu überprüfen, ob sowohl in den Augenbewegungsdaten, als auch in dem betrachteten Pixel des Filmes, grundsätzlich Bewegung vorhanden ist. Vielmehr muss für eine Klassifikation der weichen Augenfolgebewegungen zusätzlich verifiziert werden, ob die Bewegungen des Auges und die des sich bewegenden Objektes im Film synchron sind. Die Forschung im Bereich der Klassifikation von weichen Augenfolgebewegungen würde dadurch einen weiteren Schritt vorangebracht werden.

# Literatur

[Bar00] Erhardt Barth, *The minors of the structure tensor*, DAGM-Symposium (Gerald Sommer, Norbert Krüger, and Christian Perwass, eds.), Informatik Aktuell, Springer, 2000, pp. 221–228.

[BF00] Erhardt Barth and Mario Ferraro, *On the geometric structure of spatio-temporal patterns*, AFPAC (Gerald Sommer and Yehoshua Y. Zeevi, eds.), Lecture Notes in Computer Science, vol. 1888, Springer, 2000, pp. 134–143.

[Duc00] Andrew Duchowski, *Eye tracking methodology: Theory and practice*, ch. Eye-Based Interaction in Graphical Systems, Springer, 2000.

[Geg06] Karl R. Gegenfurtner, *Gehirn & wahrnehmung*, Fischer, 2006.