



METHODEN ZUR HINTERGRUNDERKENNUNG UND OBJEKTVERFOLGUNG IN BILDFOLGEN

BACHELORARBEIT
zur Erlangung des akademischen Grades
BACHELOR OF SCIENCE

Westfälische Wilhelms-Universität Münster
Fachbereich Mathematik und Informatik
Institut für Numerische und Angewandte Mathematik

Betreuung:

Prof. Dr. Martin Burger

Dr. Frank Wübbeling

Eingereicht von:

Jonas Hoppe

Münster, März 2011

Zusammenfassung

Diese Arbeit beschäftigt sich mit einigen Verfahren zur Vordergrunderkennung durch Subtraktion eines Hintergrundbildes von den Bilddaten in Bildfolgen (engl. Background Subtraction). Diese Methoden trennen den Vorder- von dem Hintergrund. Weiterhin wird auf die Personenverfolgung in Bildfolgen mittels Hintergrunderkennung eingegangen und ein Algorithmus zur Verfolgung von Zellen unter dem Mikroskop vorgestellt. Dann wird versucht Ergebnisse der Hintergrunderkennung darin einfließen zu lassen.

Eidesstattliche Erklärung

Hiermit versichere ich, *Jonas Hoppe*, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe. Gedanklich, inhaltlich oder wörtlich übernommenes habe ich durch Angabe von Herkunft und Text oder Anmerkung belegt bzw. kenntlich gemacht. Dies gilt in gleicher Weise für Bilder, Tabellen, Zeichnungen und Skizzen, die nicht von mir selbst erstellt wurden.

Die auf der DVD beigefügten Programme sind gemäß Abschnitt A.1 von mir selbst programmiert worden.

Münster, 22. März 2010

Jonas Hoppe

Inhaltsverzeichnis

1. Einleitung	1
2. Grundlagen	2
2.1. Bilder und Bildfolgen	2
3. Verfolgung von Personen in Bildfolgen	4
3.1. Grundannahmen	4
3.2. Hintergrunderkennung mittels Hintergrundbildsubtraktion	4
3.2.1. Ansatz von J. Heikkilä und O. Silvén	5
3.2.2. Abwandlungen des ersten Ansatzes	8
3.2.3. Vergleich der Verfahren	17
3.2.4. Hintergrunderkennung bei Farbbildfolgen	18
3.3. Erkennen und Verfolgen von Personen	21
4. Zellverfolgung	23
4.1. Aufgaben bei der Zellverfolgung	23
4.2. Algorithmus zur Zellverfolgung	24
4.2.1. halbautomatische Zellverfolgung	24
4.2.2. Ergebnisse	26
4.3. Einfluss von Hintergrunderkennung auf Zellverfolgung	29
4.3.1. Ansätze	29
4.3.2. Bewertung der Ergebnisse	33
5. Fazit und Ausblick	34
A. Anhang	35
A.1. Programme	35
A.2. Videos	36
Abbildungsverzeichnis	37
Literaturverzeichnis	38

1. Einleitung

In dieser Arbeit werden wir uns mit der Verfolgung von Objekten in Bildfolgen beschäftigen. Dazu wird zuerst zu unterscheiden sein, welche Bildbereiche als Vorder- und welche als Hintergrund zu zählen sind.

Wir werden insbesondere zwei Szenarien betrachten. Das sind zum einen Fußgänger und zum anderen sich bewegende Zellen unter dem Mikroskop.

Im ersten Fall ist es von Interesse die Personen automatisch zu markieren und Laufwege oder besondere Aktionen wie zum Beispiel eine Interaktion mit Gegenständen des Hintergrundes oder mit anderen Personen zu erkennen. Dies kann zum Beispiel in Überwachungsvideos genutzt werden oder zur Analyse von Fluchtwegen bei Großveranstaltungen. Als problematisch erweisen sich dabei die Umgebungen. Wechseln die Lichtverhältnisse? Welchen Einfluss haben Schatten? Was passiert, wenn sich Personen gegenseitig verdecken? Wie schafft man es genau die Objekte zu verfolgen, die von Interesse sind und nicht etwa sich bewegende Objekte des Hintergrundes wie zum Beispiel Bäume, die sich im Wind bewegen? Welche Möglichkeiten ergeben sich durch Nutzung von Farb- statt Schwarz-Weiß-Bildern? Diese Fragen und weitere Problemstellungen werden zu bearbeiten sein.

Das Verfolgen von Zellen unter dem Mikroskop, das sogenannte Celltracking, ist von Interesse, weil zum Beispiel Aktivität von Tumorzellen ein Anhaltspunkt für Ärzte sein kann eine bessere Auswahl des Therapieverfahrens zu treffen. Die Bewegungsgeschwindigkeit von Immunzellen lässt zudem darauf schließen, ob eine Infektion schnell von ihnen bekämpft werden kann.

In Abschnitt 2 wird definiert, was Bilder und Videos sind. Dann werden in Abschnitt 3 Grundannahmen der Personenverfolgung definiert und einige Methoden zur Vorder- bzw. Hintergrundbestimmung mittels Hintergrundbildsubtraktion vorgestellt. Diese werden dann anhand von Implementierungen verglichen. Danach wird in Abschnitt 3.3 ein Algorithmus zur Verfolgung von Personen in Bildfolgen basierend auf Hintergrundbildsubtraktion vorgestellt. Ein Verfahren das sich für Zellverfolgung eignet wird in Abschnitt 4 eingeführt. Abschließend wird in Abschnitt 4.3 unter Verwendung eines Programms, das dieses Verfahren durchführt, untersucht, welchen Einfluss eine vorherige Vordergrunderkennung mit den zuvor vorgestellten Algorithmen auf die Ergebnisse der Zellverfolgung haben kann.

2. Grundlagen

2.1. Bilder und Bildfolgen

Die Verfolgung von Objekten geschieht anhand von gegebenen Daten. Diese Daten müssen nun in einer handhabbaren Form vorliegen. Dazu gebe ich gemäß [3] folgende

Definition 2.1. *Ein d -dimensionales (idealisiertes) kontinuierliches Bild ist eine Funktion $u : \Omega \rightarrow \mathbb{R}$ mit dem Gebiet $\Omega \subset \mathbb{R}^d$ des Bildes, welches meist ein Rechteck oder einen d -dimensionalen Würfel darstellt.*

Ein kontinuierliches Video ist dann eine Familie $(u_t)_{t \in T}$ mit einem Intervall $T = \{t \in \mathbb{R}_0^+ : t \leq t_0\}$ für ein $t_0 \in \mathbb{R}^+$.

In dieser Arbeit werden nur zweidimensionale Bilder und Bildfolgen betrachtet. Deshalb folgt hier diese

Definition 2.2. *Ein diskretes Grauwertbild ist eine Matrix $A \in [0, 1]^{n \times m}$ für ein $n \in \mathbb{N}$ und ein $m \in \mathbb{N}$. 0 wird dabei für einen weißen Bildpunkt gesetzt und 1 für einen schwarzen. Ein Farbbild ist ein $B \in [0, 1]^{(n \times m) \times 3}$, besteht also aus drei Grauwertbildern. Dabei stellt das erste Grauwertbild die roten, das zweite die grünen und das dritte die blauen Farbanteile dar. Man spricht auch von einem rgb-Bild.*

Die einzelnen Bildpunkte von A und B werden als Pixel bezeichnet.

Eine Bildfolge oder auch ein Video oder Film ist hier eine Folge $(I_t)_{t \in T}$ mit dem diskretisierten Zeitraum $T = \{t \in \mathbb{N} | t \leq N\}$ für ein $N \in \mathbb{N}$, wobei entweder alle I_t Grauwertbilder oder Farbbilder sind. Für Farbvideos gilt dann $(I_t)_{t \in T} = (r_t, g_t, b_t)_{t \in T}$. Zwei aufeinanderfolgende Bilder haben immer den gleichen zeitlichen Abstand.

Aus einem kontinuierlichen Bild $u : [0, x_0] \times [0, y_0] \rightarrow [0, 1]$ kann eine diskretisierte Version $A \in [0, 1]^{n \times m}$ für $x_0, y_0 \in \mathbb{R}^+$ und $n, m \in \mathbb{N}$ auf folgende Weise bestimmt werden:

$$A(i, j) = \frac{\int_{(i-1) \cdot \frac{x_0}{n}}^{i \cdot \frac{x_0}{n}} \int_{(m-j) \cdot \frac{y_0}{m}}^{(m-j+1) \cdot \frac{y_0}{m}} u(x) \, dy \, dx}{\frac{x_0}{n} \cdot \frac{y_0}{m}} \quad (2.1)$$

für $i = 1, \dots, n$ und $j = 1, \dots, m$.

Umgekehrt kann aus einem diskreten Bild $A \in [0, 1]^{n \times m}$ für $x_0, y_0 \in \mathbb{R}^+$ und $n, m \in \mathbb{N}$ auf

triviale Weise ein kontinuierliches Bild $u : [0, x_0] \times [0, y_0] \rightarrow [0, 1]$ erzeugt werden, indem in einem Punkt $(x, y) \in [i \cdot x_0, (i + 1) \cdot x_0] \times [j \cdot y_0, (j + 1) \cdot y_0]$

$$u(x, y) = A(i, j) \tag{2.2}$$

gilt für alle $i = 1, \dots, n$ und $j = 1, \dots, m$.

3. Verfolgung von Personen in Bildfolgen

Bei der Verfolgung von Objekten in Bildfolgen ist es sinnvoll die einzelnen Bilder zunächst in Vorder- und Hintergrundbereiche einzuteilen, um dann die Bereiche des Vordergrundes so zusammenzufassen, dass die einzelnen Objekte voneinander unterschieden werden.

3.1. Grundannahmen

Da in Videos oder geschnittenen Filmen nicht notwendig konsistente Bedingungen vorliegen, benötigt man einige Grundannahmen, um eine Objektverfolgung durchführen zu können.

In dieser Arbeit setzen wir eine *feste Kamera*, die *nicht geschwenkt* wird, voraus. Des Weiteren ist eine *Zoomveränderung ausgeschlossen*.

Bei den vorgestellten Verfahren wird außerdem angenommen, dass sich *Objekte schnell* bewegen verglichen mit gegebenenfalls stattfindenden *langsamen Veränderungen im Hintergrund*. Auf die Geschwindigkeit von Objekten wird an späterer Stelle noch genauer eingegangen.

3.2. Hintergrunderkennung mittels Hintergrundbildsubtraktion

Die Begriffe Hinter- und Vordergrunderkennung können jeweils anstelle des anderen verwendet werden, denn die Ergebnisse sind genau komplementär zueinander. Hier werden einige Verfahren zur Hintergrunderkennung vorgestellt und anhand eines kurzen Beispielvideos mit verschiedenen Parametern getestet. Dies soll zum einen den Einfluss der Parameter zeigen und zum anderen möglichst für die Objektverfolgung weiterverwendbare Ergebnisse liefern. Es werden zu den Implementierungen die einzelnen Ergebnisse nach jeweils 62 Zeitschritten, also auch 62 Bildern, abgebildet. Dies garantiert einen Initialisierungsvorlauf, sodass die Ergebnisse nicht zu sehr vom ersten Bild abhängen.

Bei den dargestellten Tests zu den einzelnen Parametern wird jeweils nur ein Parameter verändert. Bei dem Erstellen dieser Arbeit wurden jeweils wesentlich mehr Kombinationen von Parametern überprüft. Die dargestellten Ergebnisse zeigen jeweils die Variation nur eines Parameters basierend auf den für optimal befundenen Werten.

Die erzielten Resultate hängen sehr stark von den gegebenen Daten ab, sodass die Parameter der besseren Ergebnisse nur als Richtwerte für eine Anwendung auf andere Bildfolgen

zu sehen sind. Faktoren sind hier unter anderem die Auflösung, die Bildfrequenz und die Geschwindigkeit der Objekte.

3.2.1. Ansatz von J. Heikkilä und O. Silvén

In [7] verfolgt man den naheliegendsten Ansatz. $(I_t)_{t \in T}$ sei die gegebene Bildfolge. Dann bestimmt man eine Folge von Erwartungswerten $(B_t)_{t \in T}$ für den Hintergrund rekursiv mittels

$$B_{t+1} = \alpha I_t + (1 - \alpha) B_t \quad (3.1)$$

Als B_1 setzt man I_1 . Der Aktualisierungsparameter $\alpha \in (0, 1)$ ist klein zu wählen, da sonst Objekte des Vordergrundes zu schnell in den Hintergrund übernommen werden, was dort zu falsch erkannten Vordergrundregionen führt, wo sich ein Objekt wegbewegt hat. Das betrifft vor allem relativ große und relativ langsame Objekte. Ein kleines α wäre zum Beispiel 0.1. Als Vordergrund werden nun die Pixel markiert, die

$$|I_t - B_t| > \tau \quad (3.2)$$

erfüllen. $\tau \in (0, 1)$ ist ein vordefinierter Schwellenwert.

Die Implementierung des Algorithmus mit den Parametern $\alpha = 0.07$ und $\tau = 0.13$ ergibt das in Abb. 3.1 gegebene Ergebnis.

Nun wird der Einfluss der Parameter α und τ auf das Ergebnis überprüft.

Ein Austesten verschiedener Werte für α erzeugt Abb. 3.2.

Als Schwellenwert ist hier in allen Fällen $\tau = 0.13$ gewählt. In a ist für α der Wert 0.07, in b 0.1, in c 0.2 und in d 0.3 gewählt.

Es ist zu erkennen, dass mit steigendem α die Schlieren zwischen den Personen weniger werden. Während die Personen aneinander vorbeilaufen, ist über einen längeren Zeitraum hinweg der Bereich in der Mitte verdeckt. Dies führt dazu, dass die Grauwerte der Personen stark in den Hintergrund eingehen. Da die Personen sich danach von dort entfernen, ist die Differenz zwischen Hintergrundbild und Videobild wieder größer als der Schwellenwert, obwohl sich dort keine Personen befinden. Gleichzeitig werden aber bei steigendem α die Vordergrundbereiche innerhalb der Personen weniger, da der Hintergrund immer schneller aktualisiert wird.

Verschiedene Werte für τ liefern die in Abb. 3.3 dargestellten Ergebnisse.

In allen vier Fällen wird $\alpha = 0.07$ verwendet. Für τ wird in a 0.1, in b 0.13, in c 0.2 und in d 0.3 gewählt.

Je größer τ wird, desto weniger Bereiche werden zum Vordergrund gezählt. In a werden sehr große Bereiche falsch als Vordergrund erkannt, die sich außerhalb von Personen befinden. In b, c und d werden diese immer weniger, allerdings sind auch immer weniger Anteile der

Personen dem Vordergrund zugehörig.

Insgesamt liefert dieses Verfahren noch keine gut verwendbaren Ergebnisse.

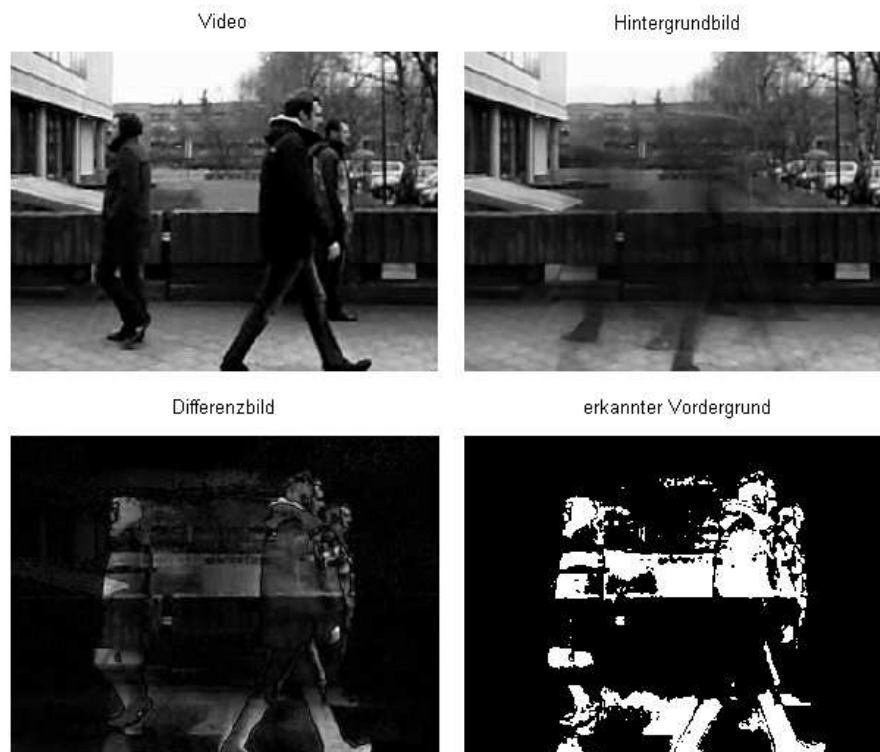


Abbildung 3.1.: Ausgabe des Programms nach dem Algorithmus von Heikkilä und Silvén

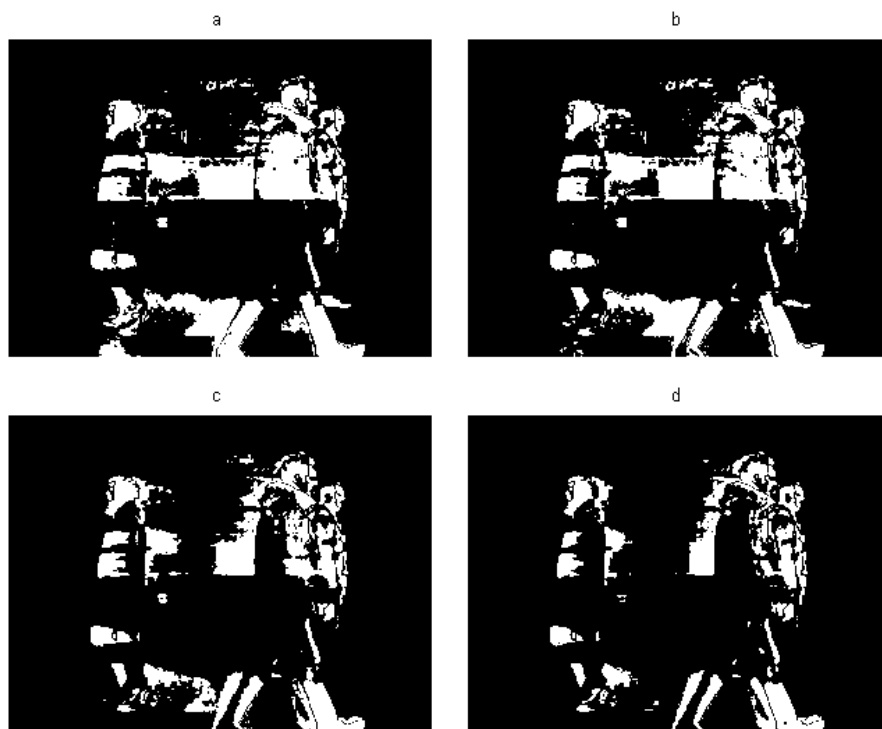


Abbildung 3.2.: Testreihe für die Wahl von α im Algorithmus von Heikkilä und Silvén

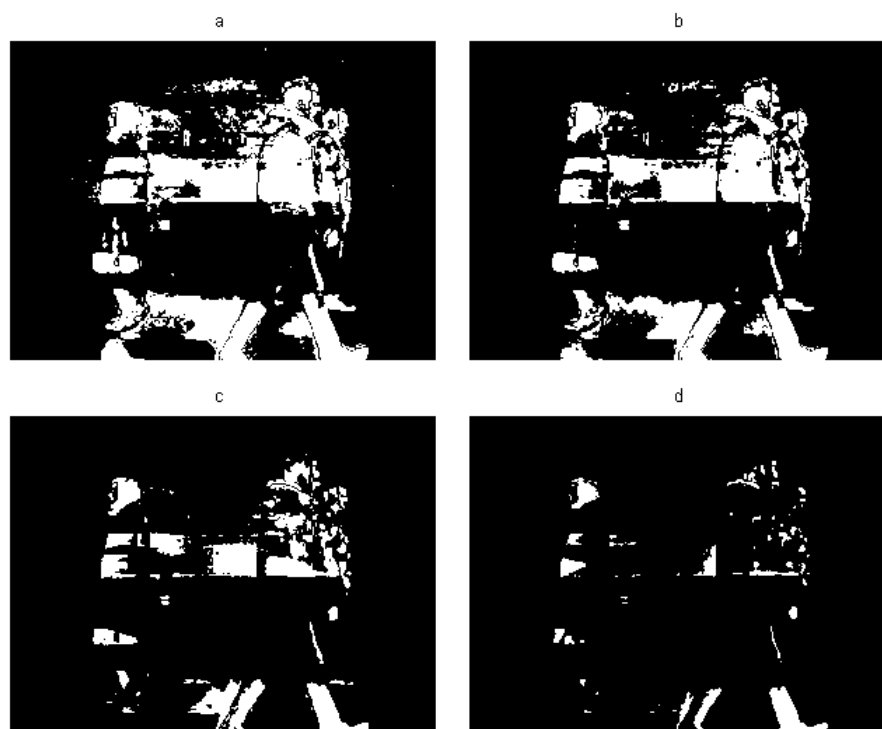


Abbildung 3.3.: Testreihe für die Wahl von τ im Algorithmus von Heikkilä und Silvén

3.2.2. Abwandlungen des ersten Ansatzes

In Hinblick auf die Verfolgung von Objekten bzw. Personen in Bildfolgen sind die im ersten Ansatz erzielten Ergebnisse noch nicht unbedingt brauchbar. Möchte man mit den nun erzielten Vordergrunddaten einzelne Personen erkennen, so sind sowohl im Hintergrund als auch im Vordergrund bei allen Ergebnissen viele Bereiche falsch zugeordnet.

Variationen liefern [2] in Bezug auf den Erwartungswert für den Hintergrund und [9] bezüglich des Schwellenwertes.

Verbesserung des Hintergrundbildes

Bei dem oben dargestellten Verfahren von Heikkilä und Silvén wird der Hintergrund in jedem Pixel gleich aktualisiert. Dabei wird die Information, ob ein Pixel dem Vorder- oder Hintergrund zugeordnet wird, nicht verwendet. In [2] wird genau diese Unterscheidung vorgenommen.

Dabei erfahren dem Hintergrund zugehörige Pixel die gleiche Aktualisierung wie zuvor. Dort, wo jedoch Vordergrund festgestellt wurde, wird mit einem im Vergleich zu α kleineren Parameter $\beta \in (0, 1)$ aktualisiert, also gilt $0 < \beta < \alpha < 1$. Damit ergibt sich für Hintergrundpixel des Zeitpunktes t

$$B_{t+1} = \alpha I_t + (1 - \alpha) B_t \quad (3.3)$$

und für Vordergrundpixel dieses Zeitpunktes

$$B_{t+1} = \beta I_t + (1 - \beta) B_t \quad (3.4)$$

Die Vordergrunderkennung erfolgt genau wie zuvor.

Auch hier werden verschiedene Werte für die Parameter α und τ getestet.

Für α liefert dies die in Abb. 3.4 abgebildeten Ergebnisse.

Es wurden für α in a 0.05, in b 0.07 in c 0.1 und in d 0.2 gewählt, jeweils mit $\tau = 0.2$.

Je größer α wird, desto weniger Artefakte bleiben an den Orten übrig, wo sich zuvor für eine längere Zeit Personen aufgehalten haben, wie am rechten Rand zu beobachten ist. Gleichzeitig tauchen mit größerem α die aus Abschnitt 3.2.1 bekannten Schlieren hinter den Personen wieder auf. Hier wird zu schnell aktualisiert.

Ein Austesten verschiedener Werte für τ liefert die in Abb. 3.5 abgebildeten Ergebnisse.

Für τ wurden in a 0.15, in b 0.2, in c 0.25 und in d 0.3 gewählt, während jeweils $\alpha = 0.1$ gesetzt wurde.

Es sind die gleichen Effekte zu beobachten wie zuvor. Für steigendes τ tauchen immer weniger unerwünschte Artefakte auf, aber auch die Vordergrundbereiche an den Stellen, wo sich Personen befinden, werden weniger.

Gegenüber dem einfachen Algorithmus stellen die hier erkannten Hintergründe ein besseres Ergebnis dar. Allerdings werden die Bereiche in der Mitte des Bildes, in denen sich Vorder- und Hintergrund vom Grauwert her nicht allzu sehr unterscheiden, immer noch nicht zum Vordergrund gezählt.

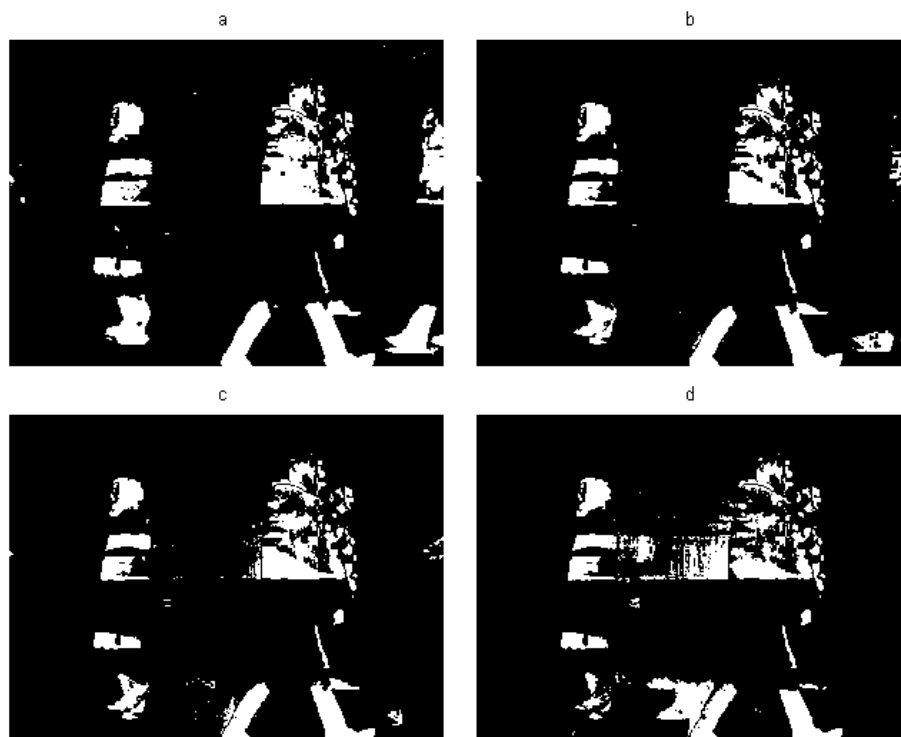


Abbildung 3.4.: Testreihe für die Wahl von α im Algorithmus mit verbesserter Hintergrundbildbestimmung

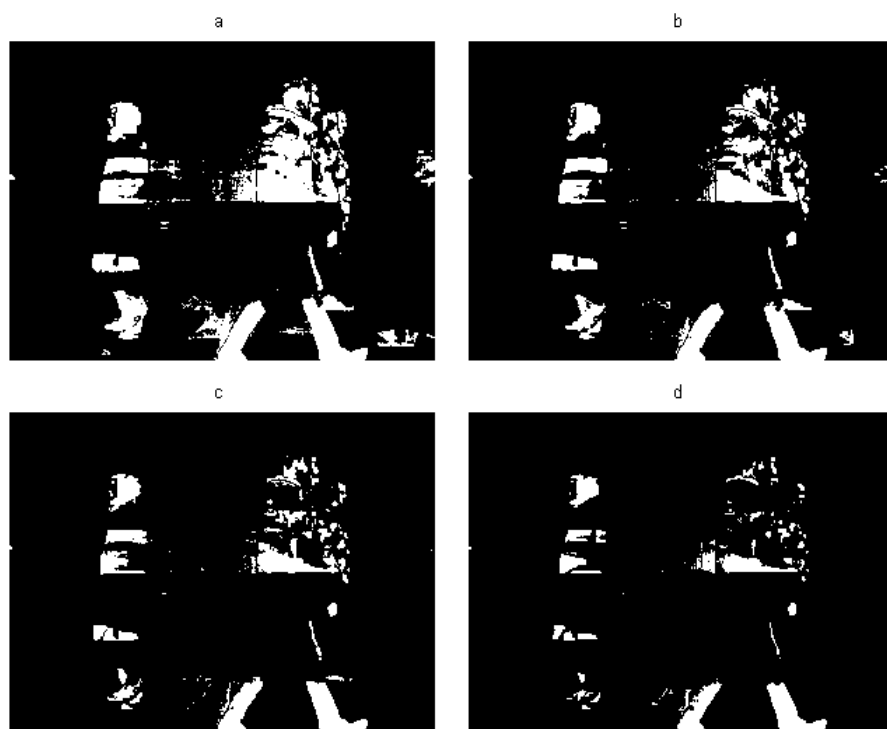


Abbildung 3.5.: Testreihe für die Wahl von τ im Algorithmus mit verbesserter Hintergrundbildbestimmung

Anpassung des Schwellenwertes

In [9] wird die Idee verwendet anstelle des fest vorgegebenen Schwellenwertes ein Vielfaches der Standardabweichung zur Abgrenzung von Vorder- und Hintergrund zu benutzen. Dazu wird nach der Berechnung von B_{t+1} wie in Abschnitt 3.2.1 ähnlich zum Erwartungswert die Standardabweichung aktualisiert.

$$\sigma_{t+1} = \sqrt{\alpha(I_{t+1} - B_{t+1})^2 + (1 - \alpha)(\sigma_t^2 + (B_{t+1} - B_t)^2)} \quad (3.5)$$

Da so gut wie jede Aufnahme mit Rauschen belegt ist, muss eine untere Grenze $\sigma_0 \in (0, 1)$ für die Standardabweichung festgelegt werden, die größer ist als die durch das Rauschen verursachte Varianz. Es wird $\sigma_1 = \sigma_0$ gesetzt.

Der Vordergrund wird nun auf folgende Weise definiert:

$$|I_t - B_t| > \lambda \cdot \max(\sigma_t, \sigma_0) \quad (3.6)$$

Dabei ist $\lambda \in \mathbb{R}$ der Faktor, der angibt, um das Wievielfache die Differenz aus Videobild und Hintergrundbild die Standardabweichung übertreffen muss, damit ein Pixel als Vordergrund gesetzt wird.

Nun sind die Parameter α , λ und σ_0 zu variieren.

Für α liefert dies die in Abb. 3.6 dargestellten Ergebnisse.

Mit $\lambda = 1$ und $\sigma_0 = 0.02$ werden für α in a 0.07, in b 0.1, in c 0.15 und in d 0.2 verwendet. Die Wahl von α hat die gleichen Effekte wie zuvor. Für große α werden nur Bereiche in der Nähe von Objektkanten als Vordergrund erkannt.

Der Test für λ liefert das in Abb. 3.7 gegebene Ergebnis.

Mit $\alpha = 0.1$ und $\sigma_0 = 0.02$ werden für λ in a 0.7, in b 1, in c 1.3 und in d 3 verwendet.

λ hat einen ähnlichen Einfluss wie zuvor τ , nur dass hier eine Abstufung nach Aktivität in den einzelnen Pixeln vorliegt. Die Ergebnisse für $\lambda = 3$ werden angefügt, weil dies dem Literaturwert entspricht. Allerdings ergeben sich damit hier keine direkt zur Weiterverwendung in der Objektverfolgung geeigneten Resultate.

Der Test für σ_0 liefert das in Abb. 3.8 gegebene Ergebnis.

Mit $\alpha = 0.1$ und $\lambda = 1$ werden für σ_0 in a 0, in b 0.01, in c 0.02 und in d 0.07 verwendet.

Da in a $\sigma_0 = 0$ gewählt wurde, ist der minimale Schwellenwert also nicht mehr vorhanden. Dies führt dazu, dass das im Video vorhanden Rauschen die Ergebnisse überall da verfälscht, wo schon lange kein richtiges, also nicht fälschlicherweise als solches erkanntes, Vordergrundobjekt gewesen ist, was besonders am oberen und rechten Rand deutlich zu sehen ist. Ein kleines σ_0 wie in b und c zu sehen, verhindert diesen Effekt. Ein zu großer Wert wie in d

lässt dort zu viele Vordergrundbereiche verschwinden, wo sich tatsächlich Personen aufhalten.

Wie schon die andere Abwandlung zeigt auch die Schwellenwertbildung über die Varianz eine Verbesserung der Ergebnisse.

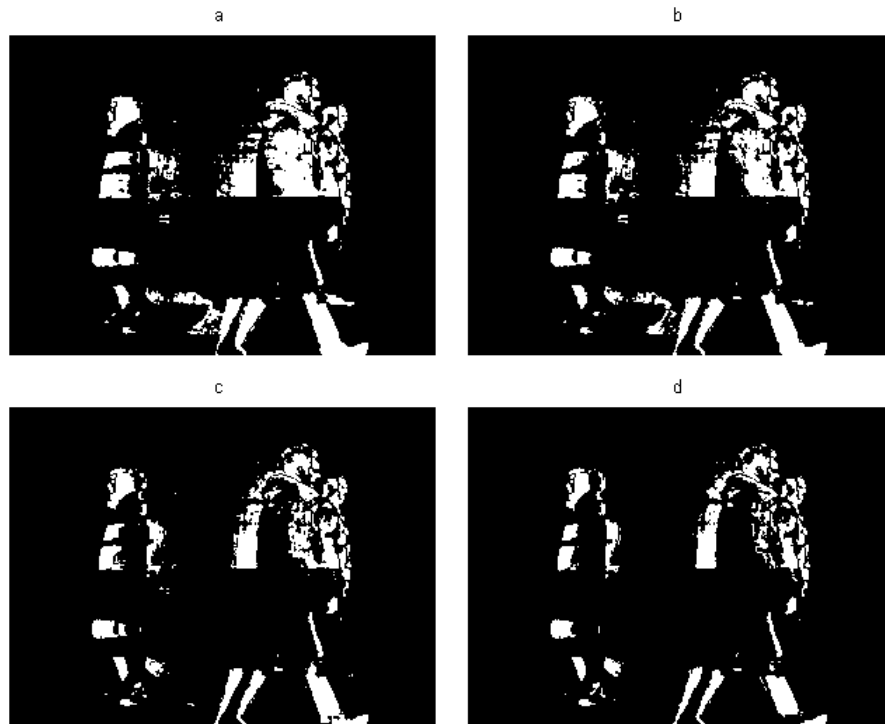


Abbildung 3.6.: Testreihe für die Wahl von α im Algorithmus mit angepasstem Schwellenwert

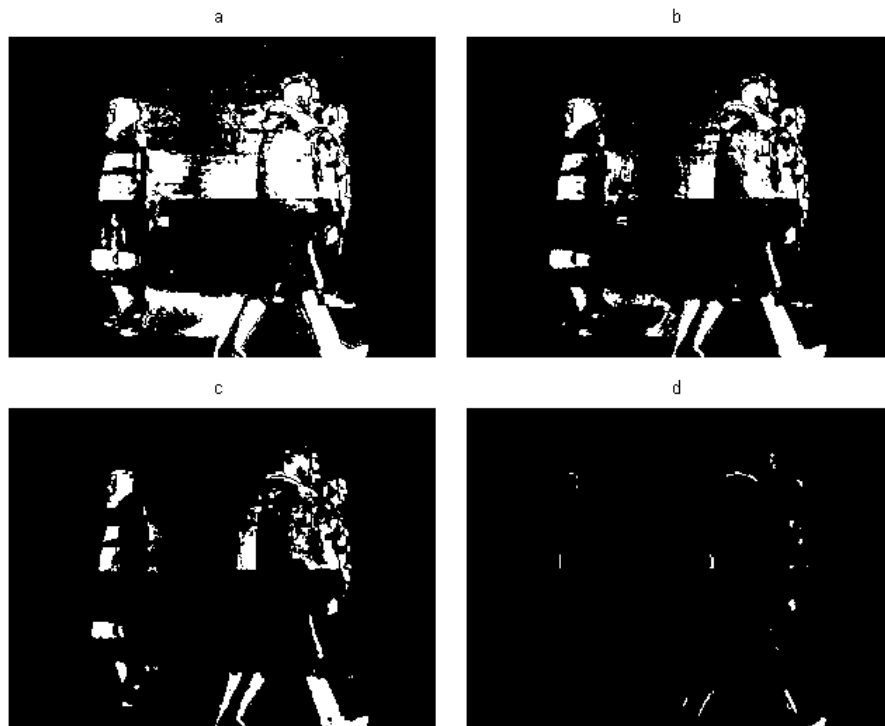


Abbildung 3.7.: Testreihe für die Wahl von λ im Algorithmus mit angepasstem Schwellenwert

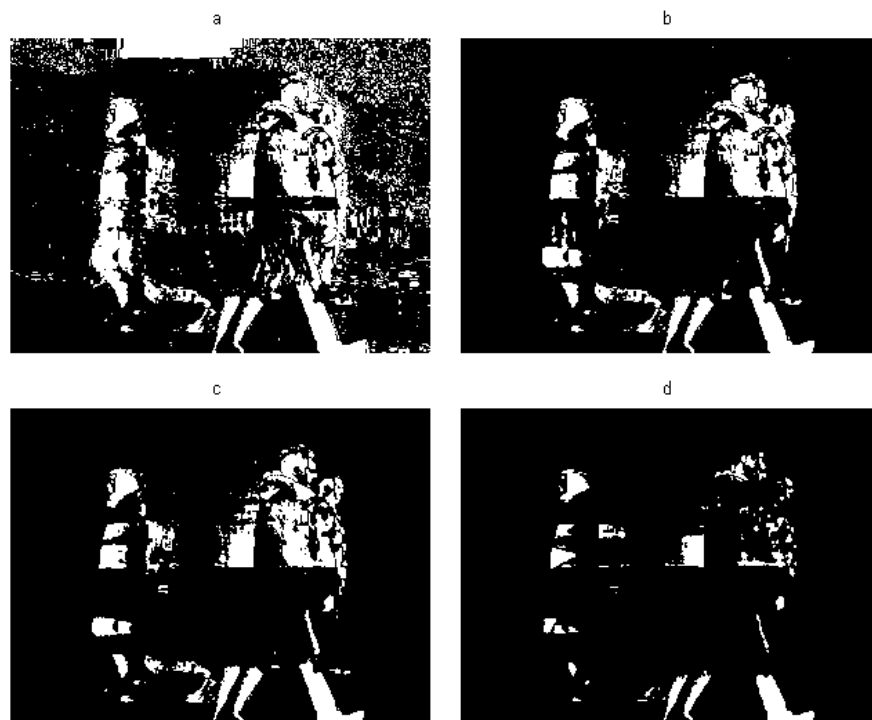


Abbildung 3.8.: Testreihe für die Wahl von σ_0 im Algorithmus mit angepasstem Schwellenwert

Kombination beider Abwandlungen

Kombiniert man nun die beiden Abwandlungen des ersten Ansatzes, so sind wieder gute Ergebnisse anhand der Parameter α , λ und σ_0 zu suchen. In Abb. 3.9 sind die Ergebnisse mit $\lambda = 0.9$ und $\sigma_0 = 0.005$ abgebildet. Dabei werden für α in a 0.07, in b 0.1, in c 0.12 und in d 0.15 verwendet.

Für den kleinsten Wert von α in a ergeben sich hier die besten Ergebnisse. Ohne so auffällige Schlieren zu verursachen wie zuvor geschehen werden die Personen zu einem großen Anteil ganzflächig erkannt.

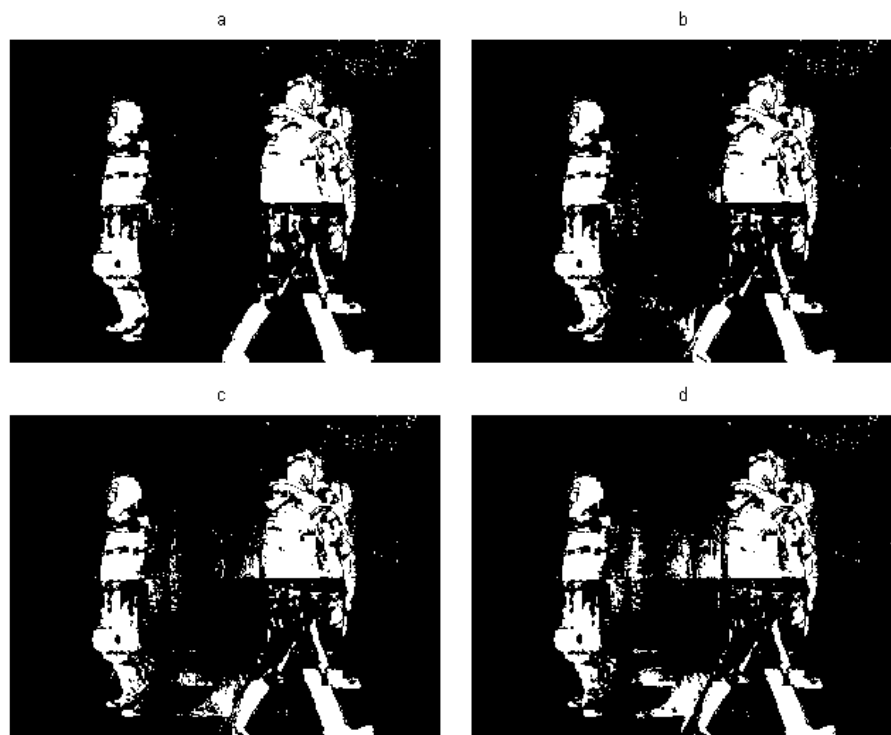
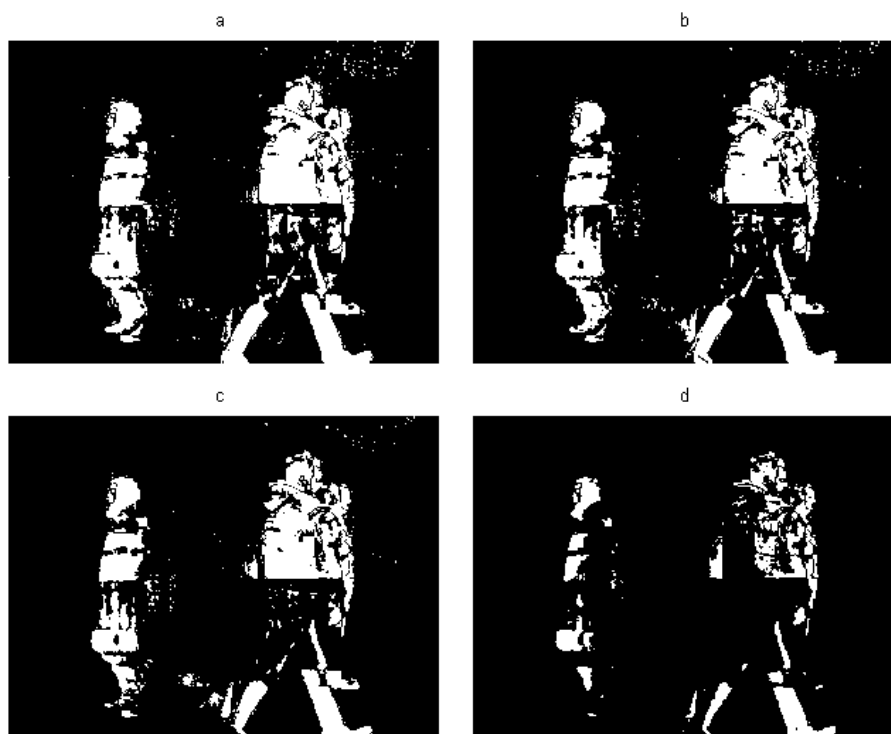
In Abb. 3.9 sind die Ergebnisse mit $\alpha = 0.1$ und $\sigma_0 = 0.005$ abgebildet. Dabei werden für λ in a 0.8, in b 0.9, in c 1 und in d 1.5 verwendet.

Für kleinere λ werden mehr Anteile der Personen erkannt, aber auch Artefakte werden mehr und größer.

In Abb. 3.9 sind die Ergebnisse mit $\alpha = 0.1$ und $\lambda = 0.9$ abgebildet. Dabei werden für σ_0 in a 0, in b 0.002, in c 0.005 und in d 0.007 verwendet.

Mit steigendem σ_0 wird das Rauschen verringert. In der Bildmitte lässt es jedoch dort Vordergrundbereiche verschwinden, wo der Hintergrund einen ähnlichen Grauwert hat wie die Objekte.

Insgesamt liefert dieses Verfahren schon relativ gute Ergebnisse, auch wenn nach wie vor bei der Parameterwahl Kompromisse zu schließen sind. Aber auch hier werden Teile der Beine der Personen nicht als Vordergrund erkannt.

Abbildung 3.9.: Testreihe für die Wahl von α im kombinierten VerfahrenAbbildung 3.10.: Testreihe für die Wahl von λ im kombinierten Verfahren

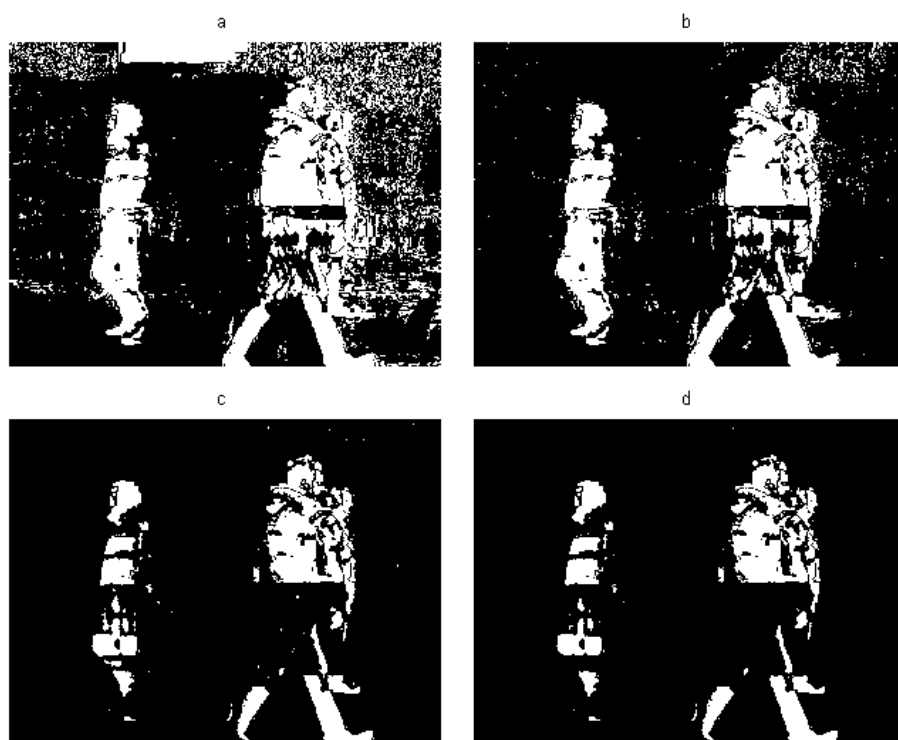


Abbildung 3.11.: Testreihe für die Wahl von σ_0 im kombinierten Verfahren

3.2.3. Vergleich der Verfahren

Ein Vergleich der optimalen Ergebnisse der einzelnen hier vorgestellten Verfahren ist in Abb. 3.12 zu sehen.

Dabei zeigt a das Verfahren nach Abschnitt 3.2.1, b gemäß Abschnitt 3.2.2, c gemäß Abschnitt 3.2.2 und d das kombinierte Verfahren.

Es ist zu erkennen, dass die differenzierte Aktualisierung des Hintergrundbildes die Schlieren verschwinden lässt. Das Verfahren mit dem über die Varianz ermittelten Schwellenwert bewirkt, dass mehr Anteile der Personen als Vordergrund erkannt werden. Das kombinierte Verfahren vereint genau diese Vorteile. Allerdings ist es auch hier problematisch in Bereichen mit einem geringen Unterschied der Grauwerte zwischen Vorder- und Hintergrund die Vordergrundbereiche zu erkennen. Der Grauwert der Hosen der Personen ist ähnlich dem Grauwert der Mauer. In allen Verfahren werden deshalb Teile der Beine nicht als Vordergrund erkannt. Im kombinierten Verfahren ist diese Eigenschaft jedoch am schwächsten ausgeprägt.

Insgesamt ist nun also ein Verfahren gefunden worden, dessen Resultate sich gut für die Personenverfolgung eignen. Dabei treten kaum Artefakte auf und es wird zum Teil sogar in Bereichen Vordergrund erkannt, in denen sich ein Vordergrundobjekt nur schwach vom Hintergrund abhebt.

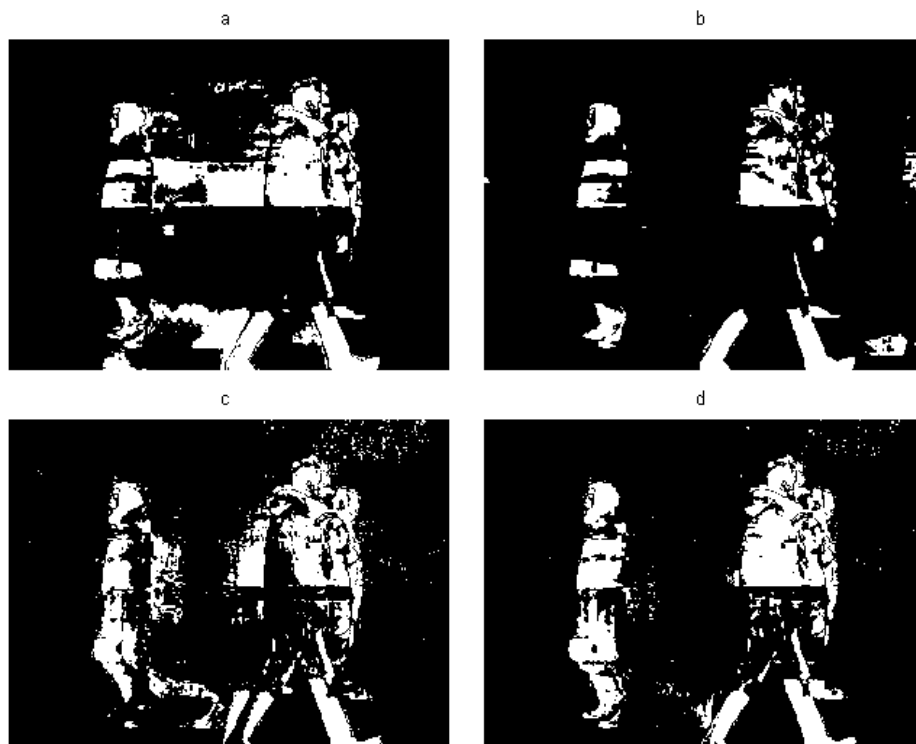


Abbildung 3.12.: Vergleich der optimalen Ergebnisse der in Abschnitt 3.2 beschriebenen Verfahren

3.2.4. Hintergrunderkennung bei Farbbildfolgen

Naheliegender scheint die Idee zu sein obige Verfahren auf drei einzelne Grauwertbildfolgen anzuwenden und dort Vordergrund zu definieren, wo in mindestens einem der drei Bilder auf Vordergrund entschieden wurde. Neben diesem durchaus sinnvollen Ansatz gibt es aber weitere Möglichkeiten die zusätzliche Information gegenüber einer einfachen Grauwertbildfolge zu nutzen. Dies geschieht zum Beispiel in den hier dargestellten Verfahren aus [9].

Farbart

Um bei Lichtveränderungen, die etwa durch Schatten von Objekten oder an- oder ausgeschaltete Lichtquellen verursacht werden, bessere Ergebnisse zu erzielen, bestimmt man in jedem Pixel die Farbart (r_{ct}, g_{ct}, b_{ct}) , auch Farbort genannt. Dies geschieht, indem durch die Helligkeit dividiert wird. Für alle $t \in T$ gilt dann:

$$r_{ct} = \frac{r_t}{r_t + g_t + b_t + \epsilon} \quad g_{ct} = \frac{g_t}{r_t + g_t + b_t + \epsilon} \quad b_{ct} = \frac{b_t}{r_t + g_t + b_t + \epsilon} \quad (3.7)$$

$\epsilon > 0$ ist sehr klein zu wählen und verhindert lediglich das Teilen durch 0. Ein Bild, das die Farbart eines Bildes darstellt, ist zum Beispiel Abb. 3.13. Nun ist auf $(r_{ct})_{t \in T}$ und $(g_{ct})_{t \in T}$ eines der obigen Verfahren anzuwenden. $(b_{ct})_{t \in T}$ kann vernachlässigt werden, da immer auf mindestens zwei von den drei Bildern Vordergrund erkannt wird, wenn solcher vorhanden ist.

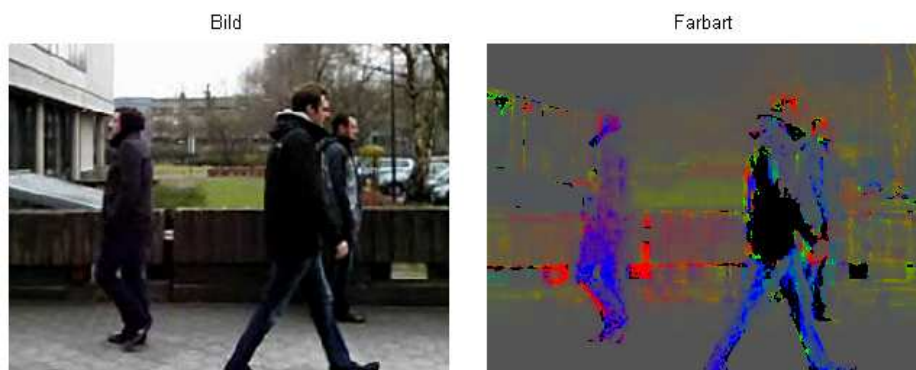


Abbildung 3.13.: Farbart

Gradientenbetrachtung

Es kann vorkommen, dass ein Objekt, das verfolgt werden soll, die gleiche Farbart hat wie der Teil des Hintergrundes, vor dem es sich bewegt. Das könnte zum Beispiel eine in der Farbe des Hintergrundes gekleidete Person sein. Hier sind die zuvor herausgerechneten Schatten wieder

hilfreich, denn Schattierungen auf Objekten bedeuten in einem Pixel einen wegen der Vergangenheit nicht zu erwartende Gradienten. Innerhalb eines Schattens, der auf einen Teil des Hintergrundes fällt, liegt zwar eine andere Helligkeit vor, was aber keine große Veränderung des Gradienten zur Folge hat. Lediglich an scharfen Rändern von Schatten, die meistens nahe bei Objekten auftreten, erscheinen unerwartete Gradienten. Somit rechtfertigt sich diese Erweiterung des Verfahrens mit der Farbart.

Die Richtungsableitungen werden mit dem Sobeloperator s_x in x-Richtung und s_y in y-Richtung bestimmt.

$$s_x = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \quad s_y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \quad (3.8)$$

Dies führt zu folgenden Richtungsableitungsbildern:

$$\begin{aligned} r_{xt} &= s_x * r_t & g_{xt} &= s_x * g_t & b_{xt} &= s_x * b_t \\ r_{yt} &= s_y * r_t & g_{yt} &= s_y * g_t & b_{yt} &= s_y * b_t \end{aligned} \quad (3.9)$$

Die Operation $*$ bezeichnet dabei die Faltung mit Dirichlet-Randbedingung. Betrachtet man Richtungsableitungen und Gradienten wieder als farbiges Bild mit der Skalierung bezüglich des Maximums aller Einträge auf 1 und des Minimums aller Einträge auf 0, da diese durchaus größer als 1 und kleiner als 0 sein können, ergibt sich Abb. 3.14. Nun werden analog zum Verfahren in Abschnitt 3.2.1 Erwartungswerte bestimmt, hier jedoch für die Richtungsableitungen anstelle der Bilddaten. Dies führt zu $(\mu_{r_{xt}}, \mu_{r_{yt}})_{t \in T}$, $(\mu_{g_{xt}}, \mu_{g_{yt}})_{t \in T}$ und $(\mu_{b_{xt}}, \mu_{b_{yt}})_{t \in T}$. Wie in Abschnitt 3.2.2 für die Bilddaten werden nun die Standardabweichungen σ_{grt} , σ_{ggt} und σ_{gbt} für den Betrag der Gradienten bestimmt. Daraus errechnen sich für jeden Zeitpunkt t durchschnittliche Standardabweichungen $\overline{\sigma_{grt}}$, $\overline{\sigma_{ggt}}$ und $\overline{\sigma_{gbt}}$ in den Bildern.

Vordergrund wird nun dort gesetzt, wo eine der folgenden Bedingungen erfüllt ist.

$$\begin{aligned} \sqrt{(r_{xt} - \mu_{r_{xt}})^2 + (r_{yt} - \mu_{r_{yt}})^2} &> \lambda \cdot \max\{\sigma_{grt}, \overline{\sigma_{grt}}\} \\ \sqrt{(g_{xt} - \mu_{g_{xt}})^2 + (g_{yt} - \mu_{g_{yt}})^2} &> \lambda \cdot \max\{\sigma_{ggt}, \overline{\sigma_{ggt}}\} \\ \sqrt{(b_{xt} - \mu_{b_{xt}})^2 + (b_{yt} - \mu_{b_{yt}})^2} &> \lambda \cdot \max\{\sigma_{gbt}, \overline{\sigma_{gbt}}\} \end{aligned} \quad (3.10)$$

λ ist hier der gleiche Parameter wie in Abschnitt 3.2.2.

Das Verfahren entscheidet nun auf Vordergrund, wenn entweder aufgrund der Farbart oder durch die Betrachtung der Gradienten Vordergrund erkannt wird.



Abbildung 3.14.: Gradient und Richtungsableitungen

3.3. Erkennen und Verfolgen von Personen

Im vorherigen Kapitel wurden einige Methoden zur Hintergrunderkennung vorgestellt. In diesem Kapitel wollen wir auf die Weiterverarbeitung der Ergebnisse eingehen. Da nun bereits festgestellt wurde, wo sich Vordergrund befindet, ist der Vordergrund in einzelne Objekte zu unterteilen, damit deren Bewegungsdaten dann ausgegeben werden können. Einen Ansatz dafür liefert wieder [9].

Um hier einen Ansatz zu finden, wird vorausgesetzt, dass die Videos aus seitlicher Perspektive die Szene wiedergeben.

Aufteilung des Vordergrundes

Wir verwenden ein Verfahren zur Hintergrunderkennung unserer Wahl und erhalten eine Folge von Binärbildern, die dort den Eintrag 1 enthalten, wo im Video Vordergrund erkannt wurde, und dort den Eintrag 0 enthalten, wo im Video Hintergrund erkannt wurde.

Der Vordergrund wird dann aufgeteilt in *Regionen*, *Personen* und *Gruppen*.

Regionen sind verbundene Komponenten, die seit mindestens $T_{fr} \in \mathbb{N}$ Zeitschritten verfolgt wurden. Vermerkte Eigenschaften dieser Region sind die Umrandungsbox, die aus den minimalen und Maximalen durch die Region belegten Koordinaten in x- und y-Richtung besteht, die Maske, die die Form der Region innerhalb der Umrandungsbox beschreibt, der Zeitstempel, der das erste Erscheinen der Region darstellt, und der Verfolgungsstatus, der den bisher zurückgelegten Weg der Region protokolliert.

Personen bestehen aus einer oder mehreren Regionen. Jede Person hat ein auf Farben basierendes Erscheinungsmodell.

Gruppen enthalten eine oder mehrere Personen. Falls zwei Personen sich eine Region teilen, werden sie zu einer Gruppe zusammengefasst.

Wie funktioniert die Verfolgung?

Wird eine neue Region erkannt, so bekommt sie einen Vermerk mit ihren Eigenschaften. Regionen können sich aufteilen. Dann bekommen alle neu entstehenden Regionen den Zeitstempel und den Verfolgungsstatus der vorherigen Region. Wenn sich Regionen verbinden, was auch möglich ist, werden Zeitstempel und Verfolgungsstatus der ältesten Region übernommen. Die Identifikation funktioniert über die Umrandungsbox. Wenn sich Umrandungsboxen von Regionen zweier aufeinanderfolgender Bilder überlappen, handelt es sich um die gleiche Region. Kann eine Region nicht mehr gefunden werden, so wird ihr Eintrag gelöscht.

Um als Person gezählt zu werden, muss eine Region verschiedene Bedingungen erfüllen. Hier geht die Einschränkung der Perspektive ein. Wir setzen Schwellenwerte fest, die eine Menge von Regionen mindestens erfüllen muss, um zu einer Person zusammengefasst zu werden.

$T_{cc} \in \mathbb{N}$ ist dabei die minimale Pixelbreite gemessen an der Projektion der Region auf die x-Achse. Zueinander gehörige Regionen dürfen nicht allzu weit voneinander entfernt sein. Weiterhin muss die Person mindestens $T_{person} \in \mathbb{N}$ Pixel an Fläche belegen. Die Schwellenwerte sind abhängig von der Szenerie zu wählen. In dem in [9] verwendeten Beispiel werden $T_{cc} = 30$ und $T_{person} = 500$ gewählt.

Gruppen entstehen, wenn sich mehrere Personen überschneiden. Wenn Regionen in einer Gruppe weit genug voneinander entfernt sind oder in der Projektion auf die x-Achse eine Lücke entsteht, teilt sich eine Gruppe wieder. Dabei wird versucht Personen anhand ihres Erscheinungsmodells wiederzuerkennen. Auch Regionen, die keine Person sind, können von Personen und Gruppen abgespalten werden, wenn zum Beispiel Objekte von Personen abgelegt werden.

Für die Erscheinungsmodelle können zum Beispiel Farbhistogramme oder Gaussian mixture Modelle verwendet werden.

Eine verwertbare Information kann dann zum Beispiel die Folge der Koordinaten der Mittelpunkte der Umrandungsboxen sein. Deren graphische Darstellung zeigt die Bewegungsverläufe der Objekte und Personen im Video. Betrachtet man zudem die Auflistung aller Personen, so kann man auch Informationen über die einzelnen Personen auswerten.

4. Zellverfolgung

In diesem Kapitel wird die Verfolgung von Zellen mit dem in [10] verwendeten Algorithmus zur Verfolgung von Zellen dargestellt. Anschließend wird untersucht welchen Einfluss die in Abschnitt 3.2 beschriebene Hintergrunderkennung auf die mit diesem Zellverfolgungsverfahren erzielten Ergebnisse haben kann.

4.1. Aufgaben bei der Zellverfolgung

Die gegebenen Daten sind Videos von Zellen unter einem Mikroskop. Dabei wurden die Zellen mit einer speziellen Beleuchtungstechnik angestrahlt. Dies ergibt Phasenkontrastmikroskopievideos wie in Abb. 4.1 abgebildet.

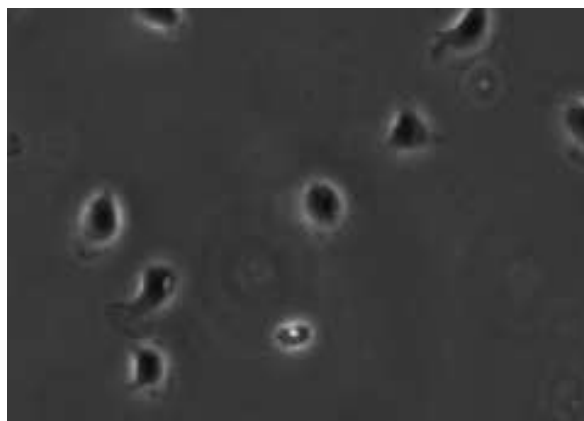


Abbildung 4.1.: Phasenkontrastmikroskopievideo

Die Aufgabe ist nun die einzelnen Zellen zu erkennen, zu allen späteren Zeitpunkten zu identifizieren und dann diese Ergebnisse auszuwerten bezüglich der Bewegung der Zellen.

Bei den Zellen im Phasenkontrastmikroskopievideo stellen sich verschiedene Probleme. Innerhalb einer Zelle können die Grauwerte variieren. Zellen können heller oder dunkler sein als der Hintergrund, sodass dies kein Kriterium für das Erkennen von Zellen ist. Zellen haben

keine klar zu definierende starre Form, sondern verformen sich während eines Videos zum Teil sehr. Die Grenze einer Zelle ist wegen der Beleuchtungstechnik nicht eindeutig zu erkennen, da sich um eine Zelle herum ein hell beleuchteter Bereich ergibt. Außerdem sind auf den Videos neben den Zellen auch ein möglicherweise inhomogener Hintergrund und eventuell Dreck, tote Zellen und Substratspuren zu sehen. Diese Bildbestandteile sollen nicht als zu verfolgende Objekte erkannt werden.

4.2. Algorithmus zur Zellverfolgung

Für das Verfahren in [10] wird angenommen, dass die Zellen in den Bildfolgen unter jeglicher Verformung ihr Volumen von einem Zeitpunkt zum nächsten nicht wesentlich verändern. Der von der Zelle abgedeckte Flächeninhalt verändert sich also nur sehr langsam.

Weiterhin wird vorausgesetzt, dass sich Zellen nicht teilen oder verbinden und keine Löcher enthalten.

4.2.1. halbautomatische Zellverfolgung

Bei der halbautomatischen Zellverfolgung findet eine minimale Interaktion mit dem Benutzer statt. Er ist angewiesen zu beobachtende Zellen im ersten Bild der eingegebenen Bildfolge zu markieren. Basierend auf dieser Eingabe erfolgt die Verfolgung der Zellen. Das Verfahren lässt sich in verschiedene Schritte aufteilen, die im folgenden genauer erläutert werden. Es werden zunächst die *Normalgeschwindigkeiten* berechnet, die ein Maß für die Aktivität in einem Pixel zu einem Zeitpunkt sind, dann findet eine Segmentierung der Normalgeschwindigkeitsbilder mit einem *Chan-Vese-Algorithmus* statt, einer Variationsmethode. Dabei ist der *Topologieerhalt* zu berücksichtigen. Dies verhindert eine große Veränderung des Volumens, plötzliche, starke Konturveränderungen, die Teilung von Zellen, das Verbinden mehrerer Zellen und Löcher innerhalb einer Zelle. Der letzte Schritt ist eine Verfeinerung der Zellkonturen mit einem Modell von *aktiven Konturen*.

Normalgeschwindigkeit

Die Normalgeschwindigkeit $(v_t)_{t \in T}$ stammt aus der Theorie des Optischen Flusses. Die Annahme ist hier, dass ein sich bewegendes Objekt in einem beobachteten Zeitraum seine Erscheinung nicht verändert. Also bleibt der Grauwert eines Objektes im Gegensatz zu seinem Ort konstant.

Dies führt für eine kontinuierliche Darstellung $u(x,t)$ eines Videos zu

$$|v_t| = \frac{|\frac{\partial}{\partial t}u(x,t)|}{|\nabla u(x,t)|_e} \quad (4.1)$$

mit $|\nabla u(x, t)|_\epsilon = \sqrt{(\partial_{x_1} u)^2 + (\partial_{x_2} u)^2 + \epsilon^2}$ für ein kleines $\epsilon > 0$, um eine Division durch 0 zu vermeiden.

Segmentierung mit einer Chan-Vese Methode

Bei der Segmentierung geht es darum in einem Bild Bereiche so zu einem Gebiet zusammenzufassen, dass es ein Objekt des Bildes überdeckt.

Dazu wird hier eine Chan-Vese Methode gemäß [12] verwendet. Die Idee ist dabei ein Bild in zwei Teile aufzuteilen. In beiden Teilen soll dabei das Bild durch einen konstanten Mittelwert approximiert werden. Eine Regularisierung findet dabei über die Länge der Kontur statt.

Wir wollen nun bei N zu verfolgenden Zellen für jede einzelne Zelle, numeriert mit $j \in \{1, \dots, N\}$, die Konstante c_{inside}^j , die den durchschnittlichen Grauwert einer Zelle darstellt, und eine Level Set Funktion $\phi_j : \Omega \rightarrow \mathbb{R}$ finden. Diese Funktionen sollen

$$\begin{aligned} \phi_j(x) &< 0, & \text{falls } x \text{ im Inneren der Zelle } j \text{ liegt,} \\ \phi_j(x) &> 0, & \text{falls } x \text{ außerhalb Zelle } j \text{ liegt,} \\ \phi_j(x) &= 0, & \text{falls } x \text{ auf dem Rand der Zelle } j \text{ liegt,} \end{aligned}$$

erfüllen. Außerdem wird die Konstante $c_{outside}$ gesucht, die den durchschnittlichen Grauwert außerhalb aller Zellen beschreibt.

Diese sollen dann das Energiefunktional

$$J = E_{data} + E_{reg} + E_{vol} \quad (4.2)$$

minimieren mit folgenden Bestandteilen:

$$\begin{aligned} E_{data}(\{\phi_j\}_{j=1, \dots, N}, \{c_{inside}^j\}_{j=1, \dots, N}, c_{outside}) &= \lambda_1 \sum_{j=1}^N \int_{\Omega} (1 - H(\phi_j))(c_{inside}^j - |v_t|)^2 dx \\ &+ \lambda_2 \int_{\Omega} \left(\prod_{j=1}^N H(\phi_j) \right) (c_{outside} - |v_t|)^2 dx \end{aligned} \quad (4.3)$$

$$E_{reg}(\{\phi_j\}_{j=1, \dots, N}) = \mu \sum_{j=1}^N \int_{\Omega} \delta(\phi) |\nabla \phi_j| dx \quad (4.4)$$

$$E_{vol}(\{\phi_j\}_{j=1, \dots, N}) = \nu \sum_{j=1}^N \left(\int_{\Omega} (1 - H(\phi_j)) dx - V_{old}^j \right)^2 \quad (4.5)$$

Dabei ist H die Heaviside-Funktion mit

$$\begin{aligned} H(\phi) &= 0, & \text{falls } \phi \leq 0, \\ H(\phi) &= 1, & \text{falls } \phi > 0, \end{aligned}$$

und δ die Dirac'sche Delta Distribution.

E_{data} ist ein Maß für den Unterschied zwischen den gegebenen Daten und den gesuchten Argumenten. E_{reg} bestraft lange Konturen. Eine Kontur ist der Bereich, in dem eine Funktion

ϕ_j den Wert 0 annehmen, also der Rand der durch $1 - H(\phi_j)$ beschriebenen Fläche. E_{vol} bestraft die Differenz des Zellvolumens der einzelnen Zellen und der Zellen aus dem vorherigen Bild der Bildfolge. V_{old}^j beschreibt hier das Volumen im vorhergehenden Bild. λ_1 , λ_2 , μ und ν sind Konstanten, die eine Gewichtung der einzelnen Terme darstellen.

Die Kontur einer Zelle j wird nun durch das Argument ϕ_j der Minimums des Funktionals J festgelegt.

Topologieerhaltung

Auch, wenn in der Chan-Vese Methode ein Strafterm für eine Veränderung des Volumens der Segmentierung einer Zelle existiert, kann es sein, dass Zellkonturen sich gegenseitig überlappen, sich verbinden, wenn sie sich berühren, oder sich teilen, wenn der Kontrast zur Umgebung zu gering ist. Deshalb wird ein Verfahren zum Topologieerhalt verwendet. Hier wird die Markierung der Zellen durch den Benutzer verwendet. Dabei wird anhand von manuell gesetzten Punkten eine Kontur um die Zellen gelegt. Damit ergibt sich für jede Zelle eine grobe Annäherung der Umrisse ohne Löcher und ohne Überschneidungen mit Konturen anderer Zellen. Eine Verfeinerung der Kontur beruhend auf der Normalgeschwindigkeit liefert eine Approximation für Form und Aufenthaltsort der Zelle. Von Bild zu Bild wird nun diese Kontur erneut angepasst, sodass die Topologie erhalten bleibt. Dies geschieht über ein in [6] dargestelltes geodätische Konturen Modell. Damit erhält man eine Abwandlung des Chan-Vese Algorithmus.

Konturenverfeinerung

In den Phasenkontrastmikroskopievideos ist zu beobachten, dass um die Zellen herum ein hell beleuchteter Bereich existiert. In den bisherigen Schritten wird dies meist als Teil der Zellen erkannt. Um nun nur die Zellen ohne den zusätzlichen beleuchteten Bereich zu umranden, muss die Konturlinie in Richtung des Zellinneren verschoben werden. Dafür wird die Idee verwendet den Laplaceoperator auf eine Gauss-geglättete Version der Bildfolge anzuwenden. Die Konturlinie wird dann so weit nach innen verlegt bis ein Maximum in den neuen Bildern erreicht ist, da hier die Grenze zwischen Zelle und Äußerem zu erwarten ist.

4.2.2. Ergebnisse

Nun wollen wir ein Beispiel betrachten. Mit dem Programm *Ctrack* segmentieren und analysieren wir dazu eine Bespielsequenz aus [11] von 5 Zellen mit einer Länge von 20 Bildern, die durch Phasenkontrastmikroskopie entstanden sind. Die Ergebnisse der Zellsegmentierung zeigt Abb. 4.2. Dabei ist oben links das Bild von den Zellen zu sehen. Rechts oben sieht man die Normalgeschwindigkeiten, links unten die manuellen Zellmarkierungen und rechts unten

das Resultat der Segmentierung.

Insbesondere wollen wir hier die Zelle in der rechten unteren Ecke der Bildfolge betrachten. Diese Zelle bewegt sich innerhalb der 20 Bilder langsam von oben nach unten. Die Analyse des Programms ist in Abb. 4.3 zu sehen. Dabei fällt auf, dass der Weg, den die Zelle zurücklegt, wie in der linken Grafik zu sehen nicht von oben nach unten verläuft, sondern einen ständigen Richtungswechsel vollzieht, ohne dass die Tendenz irgendeiner Bewegungsrichtung zu erkennen wäre. Dies hängt mit der Veränderung der Umrandung und der damit verbundenen Verschiebung des Mittelpunktes zusammen. Dass die Umrandung nicht perfekt getroffen wurde, zeigt das Bild der Zelle mit Umrandung unten rechts, in dem der obere Teil der Zelle nicht innerhalb der errechneten Umrandung liegt. Deshalb erscheint es sinnvoll zu versuchen die Hintergrunderkennung mit den Verfahren aus Abschnitt 3.2 zu nutzen, um bessere Ergebnisse zu erzielen.

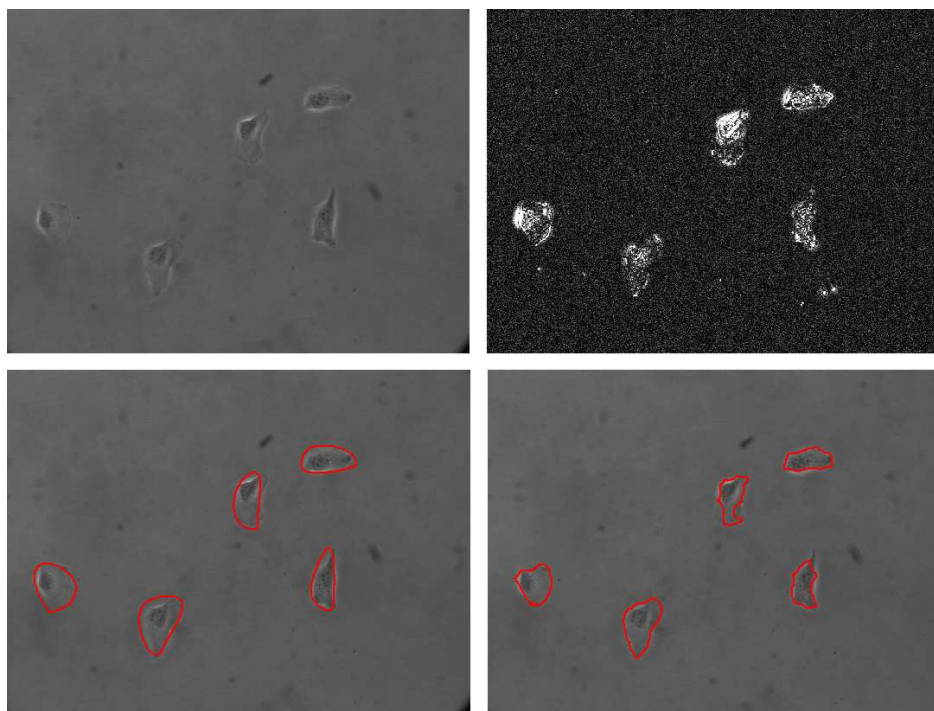


Abbildung 4.2.: Ergebnisse der Segmentierung von Zellen

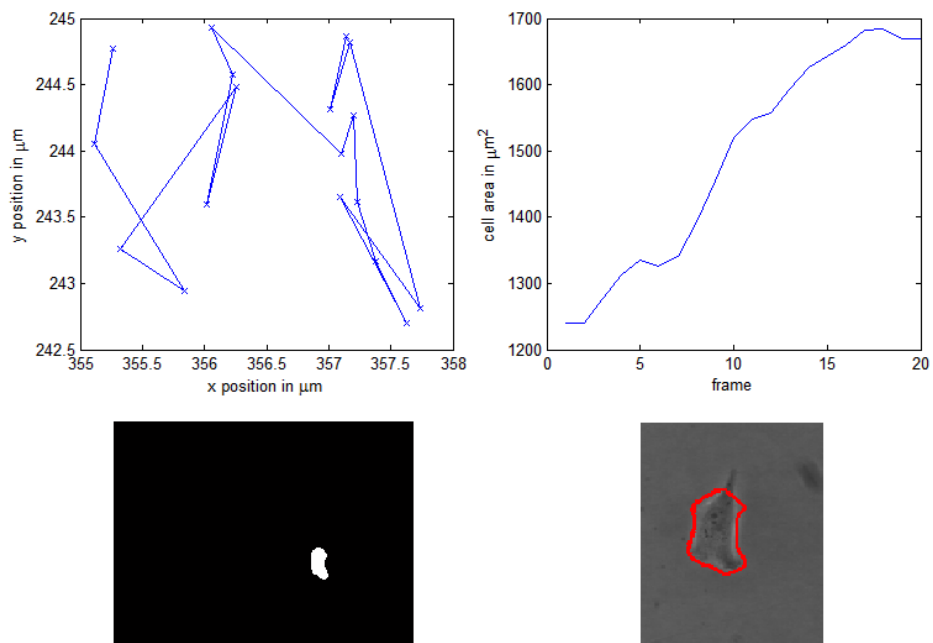


Abbildung 4.3.: Analyse der einzelnen Zelle

4.3. Einfluss von Hintergrunderkennung auf Zellverfolgung

Abschließend wird hier versucht bessere Ergebnisse bei der Zellverfolgung zu erzielen, indem das eingegebene Video der Zellen mittels Ergebnissen der Methoden aus Abschnitt 3.2 vorbearbeitet wird.

Dazu wenden wir das kombinierte Verfahren aus Abschnitt 3.2.2 auf eine Folge von 100 Bildern an, deren 20 letzte Bilder unsere Beispielsequenz aus Abschnitt 4.2.2 sind. Durch ausprobieren der Parameter ergibt sich ein geeignetes Ergebnis für $\alpha = 0.17$, $\lambda = 0.35$ und $\sigma_0 = 0.003$. Diese Werte weichen recht stark von den zuvor verwendeten ab, resultieren aber aus den schwachen Kontrasten und der relativ geringen Geschwindigkeit der Zellen. Der berechnete Vordergrund ist in Abb. 4.4 dargestellt.

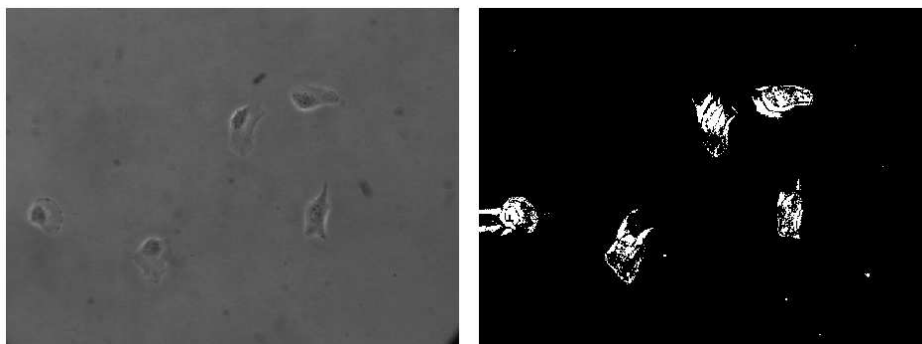


Abbildung 4.4.: Ergebnis der Hintergrunderkennung im Phasenkontrastmikroskopievideo

4.3.1. Ansätze

Es werden nun verschiedene Konzepte ausprobiert. Die Idee hinter allen hier getesteten Ansätzen ist es den Vordergrund beizubehalten und den Hintergrund zu ersetzen. Dabei betrachten wir die Binärbildfolge $(Fg_t)_{t \in T}$, die Vordergrundpixel mit 1 und den Hintergrundpixel mit 0 kennzeichnet. Sie ist das in Abb. 4.4 dargestellte Ergebnis der Hintergrunderkennung. Wir bezeichnen die neue Hintergrundbildfolge mit $(B_t^{new})_{t \in T}$ und die dadurch neu berechnete Bildfolge mit $(I_t^{new})_{t \in T}$. In einem Pixel gilt dann für jeden Zeitpunkt $t \in T$:

$$I_t^{new} = I_t \cdot Fg_t + B_t^{new} \cdot (1 - Fg_t) \quad (4.6)$$

Auf die so erhaltenen Bildfolgen $(I_t^{new})_{t \in T}$ wenden wir dann das Programm *Ctrack* an, um die berechnete Kontur der zuvor schon betrachteten Zelle und ihre Analyse mit den bereits vorgestellten Ergebnissen der Segmentierung und Verfolgung ohne die Verwendung von Hintergrunderkennung zu vergleichen.

Der Grauwert 0 als Hintergrund

Wir wählen nun für jeden Zeitpunkt $t \in T$ für B_t^{new} den Wert 0 in jedem Pixel. Somit löschen wir den alten Hintergrund. Die Anwendung des Verfahrens auf die neue Bildsequenz ergibt die in Abb. 4.5 dargestellten Analysresultate. Auch hier ist der Weg der Zelle von oben nach unten nicht nachzuvollziehen und der obere Teil der Zelle bleibt außerhalb der Umrandung.

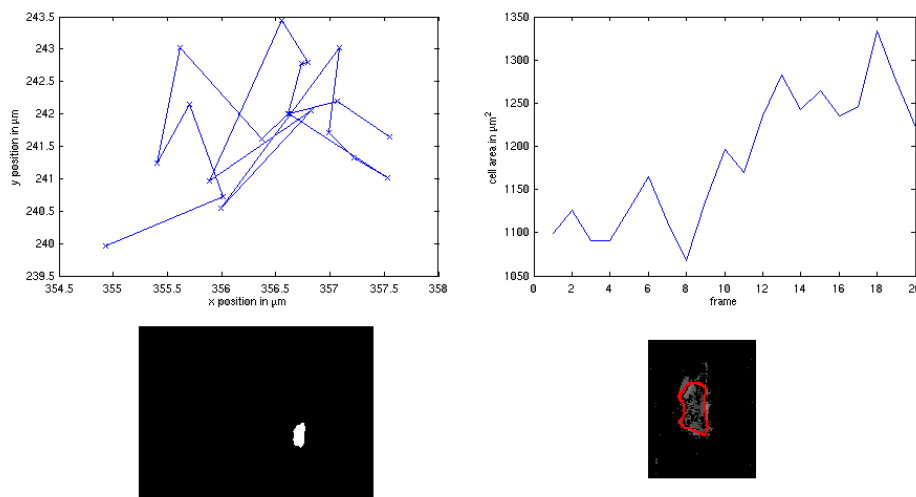


Abbildung 4.5.: Zellverfolgung mit dem Grauwert 0 als Hintergrund

Der Grauwert 1 als Hintergrund

Alternativ zum obigen Ansatz setzen den Hintergrund komplett auf 1. Doch auch dies ergibt, wie in Abb. 4.6 zu sehen ist, nicht das gewünschte Ergebnis, auch wenn festzuhalten bleibt, dass hier wie auch zuvor schon ein ganz anderer Weg festgestellt wurde als in den beiden bisherigen Versuchen, während weiterhin der obere Teil der Zelle nicht in die Umrandung aufgenommen wurde.

Durchschnittswerte der Einzelbilder als Hintergrund

Anstelle eines vorgegebenen konstanten Wertes wird nun versucht einen für die jeweiligen Bilder typischen Wert zu finden. Dazu wird hier der Mittelwert \bar{c}_t über alle Grauwerte des Bildes $I_t \in [0, 1]^{n \times m}$ eines Zeitpunktes $t \in T$ berechnet.

$$\bar{c}_t = \frac{\sum_{i=1}^n \sum_{j=1}^m I_t(i, j)}{n \cdot m} \quad (4.7)$$

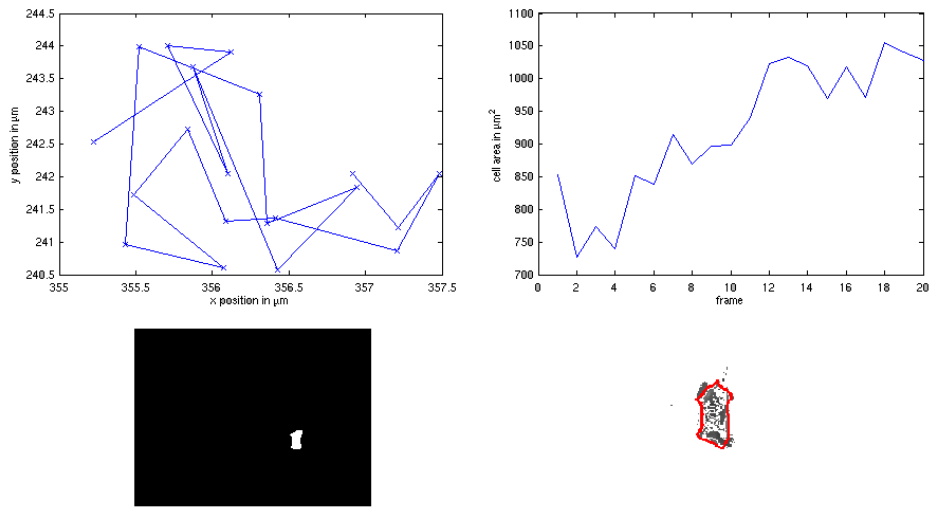


Abbildung 4.6.: Zellverfolgung mit dem Grauwert 1 als Hintergrund

Dies führt zu dem Analyseergebnis in Abb. 4.7. Darin ist tatsächlich eine Bewegungstendenz in vertikaler Richtung zu erkennen. Die horizontale Schwankung bleibt im Rahmen von drei Pixel. Außerdem ist zu beobachten, dass schon mehr obere Anteile der Zelle in den segmentierten Bereich aufgenommen wurden. Somit ist hier tatsächlich eine Verbesserung des Ergebnisses erzielt worden.

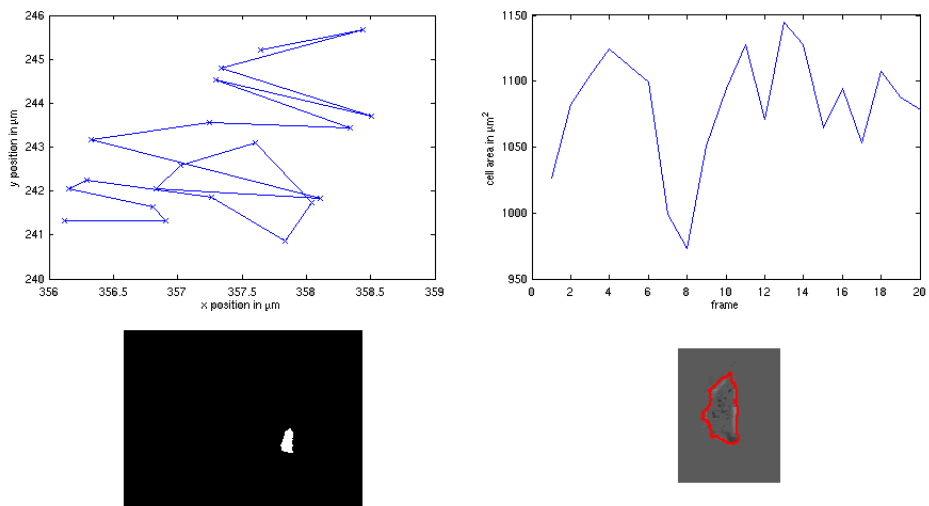


Abbildung 4.7.: Zellverfolgung mit Durchschnittswerten der Einzelbilder als Hintergrund

Durchschnittswerte des Hintergrundes der Einzelbilder als Hintergrund

Weiterhin kann nun der Ansatz verfolgt werden für den Grauwert des neuen Hintergrundes nur über dem Hintergrund zu mitteln. Dabei berechnen wir diesmal \bar{c}_t auf folgende Weise:

$$\bar{c}_t = \frac{\sum_{i=1}^n \sum_{j=1}^m I_t(i, j) \cdot (1 - Fg_t(i, j))}{\sum_{i=1}^n \sum_{j=1}^m (1 - Fg_t(i, j))} \quad (4.8)$$

Da hierbei aber fast der gleiche Wert errechnet wird wie zuvor bei der Mittelung über das ganze Bild, ergibt sich ein fast identisches Resultat, abgebildet in Abb. 4.8. Dies erfolgt wegen der geringen Varianz der Grauwerte in den Bildern und dem geringen Anteil des Vordergrundes am Bild.

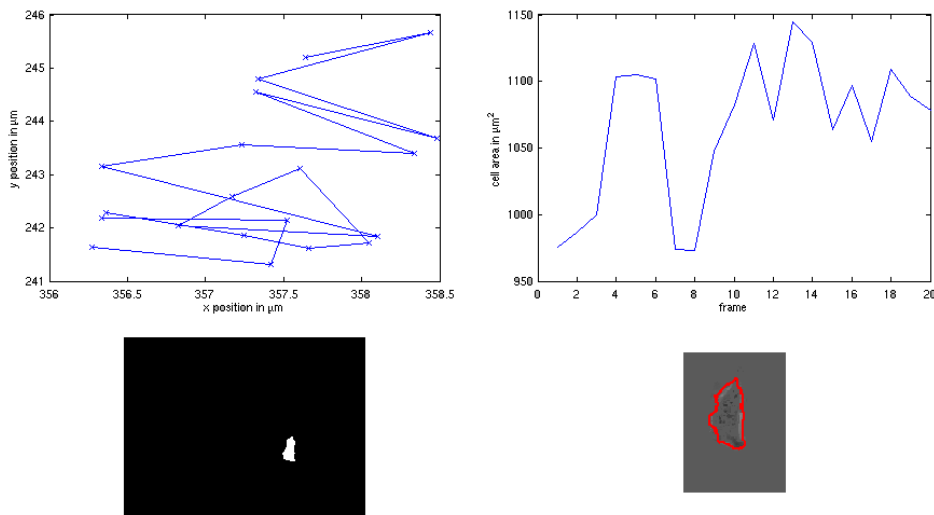


Abbildung 4.8.: Zellverfolgung mit Durchschnittswerten des Hintergrundes der Einzelbilder als Hintergrund

Geglättete Version des Videos als Hintergrund

Anstatt konstante Werte für die Hintergründe zu nehmen, folgt als letzte Idee den Hintergrund durch eine geglättete Version $(I_t^g)_{t \in T}$ des Videos zu ersetzen. Wir glätten zu die einzelnen Bilder mehrfach durch die Faltung mit einem Gaußkern.

Das Analyseresultat in Abb. 4.9 ist im Gegensatz zu den vorherigen beiden Ansätzen nicht erfreulich, auch wenn die erkannte Bewegung konsistenter zu sein scheint als ohne die Verwendung von Hintergrunderkennung.

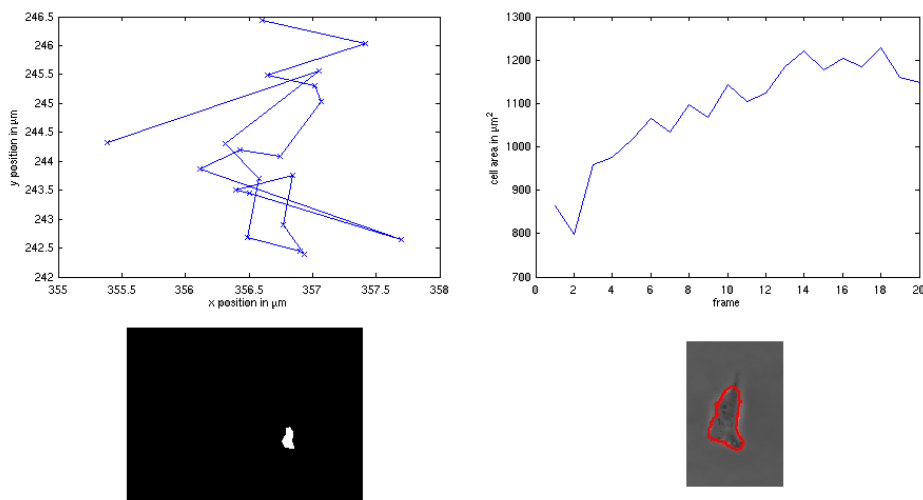


Abbildung 4.9.: Zellverfolgung mit einer geglätteten Version des Videos als Hintergrund

4.3.2. Bewertung der Ergebnisse

Die Feinheiten der Zellverfolgung konnten in dem gegebenen Beispiel verbessert werden, indem der Mittelwert der Einzelbilder als konstanter Hintergrundgrauwert verwendet wurde. Allerdings wurde hier nur ein Beispiel und keine große Testreihe verwendet, was nötig wäre, um allgemeinere Aussagen über den Einfluss von Hintergrunderkennung mittels Hintergrundbildsubtraktion auf die Zellverfolgung zu treffen. Außerdem wurde hier nur eine sehr kurze Bildsequenz bearbeitet. In einer längeren Bildfolge ist zu erwarten, dass das Resultat für den Weg die Bewegungstendenz wiedergibt, wenn auch mit starken Schwankungen wie hier gesehen. Möglicherweise ergeben sich auch bessere Ergebnisse, wenn im Programm Ctrack andere Gewichtungparameter für das Funktional aus Abschnitt 4.2.1 verwendet werden.

5. Fazit und Ausblick

Wir haben festgestellt, dass Hintergrunderkennung mittels Hintergrundbildsubtraktion sehr hilfreich sein kann bei der Objektverfolgung in Bildfolgen. Sie kann sowohl dazu eingesetzt werden der eigentlichen Objektverfolgung voranzugehen, sodass im Anschluss Vordergrundbereiche zu Objekten zusammengefasst und somit verfolgt werden können, als auch zur Verbesserung der Ergebnisse aus anderen Verfahren, indem sie zur Veränderung der Eingangsdaten verwendet wird.

Bei den in dieser Arbeit vorgestellten Methoden zur Hintergrunderkennung ist zu bemerken, dass die Wahl der vorzugebenden Parameter oft einen Kompromiss darstellt zwischen fälschlicherweise als Vordergrund und fälschlicherweise als Hintergrund erkannten Bereichen. In Echtzeit funktionierende Objektverfolgung war hier noch nicht möglich. Der Grund dafür ist der zum Teil sehr hohe Rechenaufwand der Implementierungen. Hier werden der technische Fortschritt, eine effizientere Programmierung und vor allem weiter entwickelte Verfahren zielführend sein, die zudem noch bessere Ergebnisse als die hier erzielten liefern werden.

In dieser Arbeit haben wir uns auf einige einfache Verfahren zur Hintergrunderkennung und Objektverfolgung konzentriert. Es gibt jedoch bereits viele kompliziertere Verfahren.

In [4] verfolgt man einen stochastischen Ansatz. Anhand eines Modells für die Erscheinung eines Objektes wird hier das Maximum der Wahrscheinlichkeit für dessen Position bestimmt. Ähnlich dazu wird in [1] versucht mit dem Erzeugen eines Eigenraumes eines Objektes, also der Menge aller affinen Transformationen eines Bildes davon, versucht dieses in der Bildfolge wiederzuerkennen.

Anstelle von einem einzigen Erwartungswert für jeden Pixel wie in allen in dieser Arbeit vorgestellten Verfahren zur Hintergrunderkennung verwendet werden in [5] drei bis fünf Erwartungswerte bestimmt. Mit einer adaptiven Gewichtung der einzelnen Erwartungswerte und mit der Standardabweichung von den Erwartungswerten wird hier über Vorder- und Hintergrund entschieden.

A. Anhang

A.1. Programme

Hier wird beschrieben, welche der auf der DVD beigefügten Programme von anderen Autoren stammen und welche von mir selbst programmiert wurden.

Verwendete Programme anderer Autoren

Das Programm *Ctrack* wurde von Michael Möller programmiert. Es handelt sich dabei um das in [10] vorgestellte Verfahren, dessen Funktionsweise in Abschnitt 4.2 vorgestellt wird.

Selbst erstellte Programme

Für Abschnitt 3.2 wurden folgende Programme implementiert:

- *Background_Subtraction.m*
- *heikkila.m* für Abschnitt 3.2.1
- *FrameDiff_rek.m* für Abschnitt 3.2.2
- *FrameDiff_Var.m* für Abschnitt 3.2.2
- *FrameDiff_rekVar.m* für Abschnitt 3.2.2
- *Videoeinleser.m*
- *VideoeinleserCol.m*
- *Gradient.m*
- *Farbart.m*

Für Abschnitt 4 wurden folgende Programme implementiert:

- *Zellvideo.m*
- *ZellvideoBgSub.m*

- *ZellvideosBgSubAnw.m*

Hinweise zur Verwendung der Programme

Um die Ergebnisse aus Abschnitt 3.2 nachzuvollziehen, empfiehlt sich das Programm *Background.Subtraction*.

Hier können alle Anwendungen der Verfahren, die die Abbildungen erzeugen, aufgerufen werden.

Dabei werden alle weiteren für Abschnitt 3.2 implementierten Programme verwendet.

Die Videos *mov0* für Abschnitt 4.2.2 und *mov1* bis *mov5* für Abschnitt 4.3 werden von *ZellvideosBgSubAnw.m* erzeugt. Die mit *Ctrack* erzielten Ergebnisse sind in den Dateien *Celltracking0.mat* bis *Celltracking5.mat* in gleicher Reihenfolge wie in Abschnitt 4 vorgestellt gespeichert.

A.2. Videos

Die in Abschnitt 4 verwendeten Videos von Zellen wurden zur Verfügung gestellt von Albrecht Schwab, Institut für Physiologie, WWU Münster.

Das in Abschnitt 3.2 verwendete Video *Beispiel2.avi* wurde von mir im März 2011 selbst angefertigt. Darin mitgewirkt haben Sebastian Suhr und Christoph Jansen.

Abbildungsverzeichnis

3.1. Ausgabe des Programms nach dem Algorithmus von Heikkilä und Silvén . . .	6
3.2. Testreihe für die Wahl von α im Algorithmus von Heikkilä und Silvén	7
3.3. Testreihe für die Wahl von τ im Algorithmus von Heikkilä und Silvén	7
3.4. Testreihe für die Wahl von α im Algorithmus mit verbesserter Hintergrund- bildbestimmung	9
3.5. Testreihe für die Wahl von τ im Algorithmus mit verbesserter Hintergrund- bildbestimmung	10
3.6. Testreihe für die Wahl von α im Algorithmus mit angepasstem Schwellenwert	12
3.7. Testreihe für die Wahl von λ im Algorithmus mit angepasstem Schwellenwert	13
3.8. Testreihe für die Wahl von σ_0 im Algorithmus mit angepasstem Schwellenwert	13
3.9. Testreihe für die Wahl von α im kombinierten Verfahren	15
3.10. Testreihe für die Wahl von λ im kombinierten Verfahren	15
3.11. Testreihe für die Wahl von σ_0 im kombinierten Verfahren	16
3.12. Vergleich der optimalen Ergebnisse der in Abschnitt 3.2 beschriebenen Verfahren	17
3.13. Farbart	18
3.14. Gradient und Richtungsableitungen	20
4.1. Phasenkontrastmikroskopievideo	23
4.2. Ergebnisse der Segmentierung von Zellen	27
4.3. Analyse der einzelnen Zelle	28
4.4. Ergebnis der Hintergrundbestimmung im Phasenkontrastmikroskopievideo . .	29
4.5. Zellverfolgung mit dem Grauwert 0 als Hintergrund	30
4.6. Zellverfolgung mit dem Grauwert 1 als Hintergrund	31
4.7. Zellverfolgung mit Durchschnittswerten der Einzelbilder als Hintergrund . . .	31
4.8. Zellverfolgung mit Durchschnittswerten des Hintergrundes der Einzelbilder als Hintergrund	32
4.9. Zellverfolgung mit einer geglätteten Version des Videos als Hintergrund . . .	33

Literaturverzeichnis

- [1] Michael J. Black and Allan D. Jepson. Eigenttracking: Robust matching and tracking of articulated objects using a view-based representation. *Int. J. Comput. Vision*, 26:63–84, January 1998. 34
- [2] T. E. Boult, R. Micheals, X. Gao, P. Lewis, C. Power, W. Yin, and A. Erkan. Frame-rate omnidirectional surveillance and tracking of camouflaged and occluded targets. In *Visual Surveillance, 1999. Second IEEE Workshop on, (VS'99)*, pages 48–55, 1999. 8
- [3] M. Burger and C. Brune. Variationsmethoden in der Biomedizinischen Bildgebung, oct 2010. Skript zur gleichnamigen Vorlesung. 2
- [4] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 2, pages 142 –149 vol.2, June 2000. 34
- [5] W. E. L. Grimson, C. Stauffer, R. Romano, and L. Lee. Using adaptive tracking to classify and monitor activities in a site. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR '98*, pages 22–, Washington, DC, USA, 1998. IEEE Computer Society. 34
- [6] Xiao Han, Chenyang Xu, and Jerry L. Prince. A topology preserving level set method for geometric deformable models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:755–768, 2003. 26
- [7] Janne Heikkilä and Olli Silvén. A real-time system for monitoring of cyclists and pedestrians. *Image and Vision Computing*, 22(7):563 – 570, 2004. Visual Surveillance. 5
- [8] Alan M. Mcivor. Background Subtraction Techniques. In *Proc. of Image and Vision Computing, Auckland, New Zealand, 2000.*, 2000.
- [9] S. J. Mckenna. Tracking Groups of People. *Computer Vision and Image Understanding*, pages 42–56, 2000. 8, 11, 18, 21, 22
- [10] M. Moeller, M. Burger, P. Dieterich, and A. Schwab. A framework for automated cell tracking in phase contrast microscopic videos based on normal velocities. Technical report, WWU Muenster, 2010. 23, 24, 35

-
- [11] Albrecht Schwab. Zellvideos. Institut für Physiologie, WWU Münster. 26
- [12] Luminita A. Vese and Tony F. Chan. A multiphase level set framework for image segmentation using the mumford and shah model. *International Journal of Computer Vision*, 50:271–293, 2002. 10.1023/A:1020874308076. 25