

Diplomarbeit

Verfolgung kreisförmiger Objekte in Stereobildsequenzen

Robert Borchers
1493863

Universität Bremen
Fachbereich 3
Informatik

4. Dezember 2006

Inhaltsverzeichnis

1	Einleitung	5
1.1	Gliederung der Arbeit	5
2	Grundlagen	7
2.1	Statistische Grundlagen	7
2.2	Ausgleichsrechnung	8
2.3	RANSAC	10
2.4	Houghtransformation	12
2.5	Grundlegende Operationen der Bildverarbeitung	13
2.5.1	Schwelwertverfahren	13
2.5.2	Kontrastverstärkung	14
2.5.3	Bildglättung	16
2.5.4	Kantendetektion	17
2.5.5	Eckendetektion	17
2.5.6	Labeling	19
2.5.7	Erosion, Dilatation Opening und Closing	20
2.6	Messtand	22
2.6.1	Aufbau einer CCD Kamera	22
2.6.2	Stereoskopie	23
3	Verfahren zur Ellipsoidendetektion	25
3.1	Ellipsoidendetektion nach Hough	25
3.2	Fittingverfahren	28
4	Implementierung	31
4.1	Bildvorverarbeitung	31
4.1.1	Kantenerzeugung aus einem Realbild	31
4.1.2	Filtern des Kantenbildes	32
4.2	Ellipsoidendetektion	37
4.2.1	Wahl der Verfahren	37
4.2.2	Ellipsoidenerkennung mit Houghtransformation	38
4.2.3	Ellipsoidenerkennung mit Fitting	40
4.2.4	Vergleich der Verfahren	42
4.3	Erhöhung der Erkennungsleistung	44
4.3.1	Zusammensetzen unterbrochener Linien	44
4.3.2	Binarisierung des Ausgangsbildes	46
4.3.3	Auswahl der gesuchten Ellipsen aus den Stereobildern	46
4.3.4	Zusammenfassung des Einzelraderkennungsalgorithmus	48
4.4	Radverfolgung	49
4.5	Berechnung der Tiefeninformationen	50
5	Experimente	51
5.1	Evaluation verschiedener Implementierungsdetails	51
5.1.1	Tangentenbestimmung bei der Houghtransformation	51
5.1.2	Evaluation der Erkennungsgenauigkeit	53
5.2	Evaluation der Verfahrensqualität	53
5.2.1	Robustheit bei Rauschen	54
5.2.2	Robustheit bei zusammenhängender Störung	56

5.2.3	Bestimmung des nötigen Ellipsenanteils zur korrekten Erkennung	57
5.2.4	Fazit der Evaluation an künstlichen Ellipsen	58
5.3	Evaluation der Besonderheiten von Radbildern	59
5.3.1	Fazit der Evaluation an Radbildern	61
5.4	Evaluation des Verfahrens zur Labelzusammensetzung unterbrochener Ellipsen	63
5.4.1	Test an künstlichen Ellipsen	63
5.4.2	Test an Radbildern	64
5.5	Test an einzelnen Rädern	66
5.6	Test am vorbeifahrenden Fahrzeug	70
5.6.1	Verfolgung künstlicher 3D Kreise	70
5.6.2	Verfolgung von Autorädern in Stereobildern	72
6	Zusammenfassung	74
6.1	Fazit	74
6.2	Ausblick	75
7	Anhang	77
7.1	Anteil eigener und fremder Verfahren	77

1 Einleitung

Automatische Assistenz- und Diagnosesysteme sind in der Fahrzeugreparatur mittlerweile ein unverzichtbares Hilfsmittel geworden. Heutige Systeme können mit Hilfe von Markern und Stereokamerasystemen mikrometergenaue Positionsangaben einzelner Fahrzeugelemente berechnen. Da das Aufbringen der Marker aber weiter manuelle Arbeit bedeutet, ist die autonome Erkennung der einzelnen Fahrzeugelemente ein aktuelles Forschungsgebiet in der Automobilbranche. Diese Arbeit widmet sich dabei dem Teilbereich der Suche nach den Rädern der Automobile, deren Detektion aus nachfolgenden Gründen eine große Schwierigkeit darstellt.

Da das Gewicht des Automobils auf die Räder drückt, bildet der Reifen weder einen Kreis, noch eine Ellipse. Hierdurch stellt der Übergang von Felge zu Reifen die einzige geometrische Form des Rades dar. Dies ist für Bilderkennungsverfahren eine besondere Herausforderung, da Felgen in Form und Farbe sehr individualistisch gestaltet sein können.

Verschiedene Verfahren, wie z.B. [AT04a], [AT04b] oder [LH90] haben sich mit der Raderkennung an einem Auto beschäftigt, allen liegt jedoch zugrunde, dass sie das Rad nur grob lokalisieren, während die Erkennung für weitere automatische Arbeiten am Rad mindestens pixelgenau sein muss. Zudem nutzen viele Verfahren zur Radlokalisierung eine Kreiserkennung im 2D Bild, während die projektive Abbildung eines 3D Kreises eine Ellipse darstellt. Eine Möglichkeit zur exakten Radlokalisierung ergibt sich daher durch die Suche nach Ellipsen in den Kamerabildern.

Diese Diplomarbeit beschäftigt sich daher mit der Evaluation, ob Autoräder allein durch Ellipsensuchverfahren während einer Vorbeifahrt an einem Stereokamerasystem mit mindestens pixelgenauer Exaktheit erkannt werden können. Aus der Vielzahl der existierenden Ellipsensuchverfahren (siehe Kapitel 3) wird das im Sinne der Anwendung optimale ausgewählt und auf die Raderkennung angepasst. Die Zusammensetzung der Einzelbilder zu einer 3D Positionsangabe erfolgt bei geringer Genauigkeit der erkannten Ellipsen durch eine Korrespondenzsuche im Bild, bei hoher Genauigkeit können hierfür die Ellipsenparameter genutzt werden.

1.1 Gliederung der Arbeit

Die Arbeit wurde in verschiedene Kapitel aufgeteilt, die logisch aufeinander aufbauen. Die Grundlagen für das Verständnis der verwendeten Verfahren werden in Kapitel 2 erläutert.

In Kapitel 3 wird der aktuelle Forschungsstand der Ellipsensuche erläutert, sowie zwei verschiedene Verfahrenstechniken vorgestellt.

Verschiedene Ellipsensuchverfahren werden in Kapitel 3 in ihrer Theorie erläutert, sowie in Kapitel 4 in ihrer hier umgesetzten Implementation. Dafür nötige Tests werden in Kapitel 5 durchgeführt und beschrieben. In diesem Kapitel wird auch der Erkennungsalgorithmus am Einzelrad und bei der Vorbeifahrt getestet und bewertet. Eine Zusammenfassung der hier durchgeführten Arbeiten gibt Kapitel 6, Die Auswertung des Ergebnisses findet in Kapitel 6 statt.

2 Grundlagen

2.1 Statistische Grundlagen

Algorithmen wie die Ausgleichsrechnung oder der Otsu Schwellwertoperator müssen mit großen Mengen von Messpunkten umgehen und Aussagen über diese Mengen treffen können. Dafür bedienen sie sich statistischer Merkmale wie dem Mittelwert, der Standardabweichung, der Varianz und der Kovarianz.

Der **Mittelwert** ist der durchschnittliche Wert aller Messpunkte. Sei X_i ein Messpunkt aus der Menge aller n Punkte und \bar{X} der Mittelwert, dann gilt:

$$\bar{X} = \frac{\sum_{i=1}^N X_i}{n} \quad (1)$$

Da der Mittelwert alleine nicht besonders aussagekräftig ist, weil z.B. [1 2 9 10] und [4 5 6 7] denselben Mittelwert bei unterschiedlich großem Zahlenraum haben, benutzt man eine weitere Größe, die die Streuung der Messpunkte angibt. Die **Varianz** gibt hierzu den durchschnittlichen quadratischen Abstand eines Messwertes X vom Mittelwert \bar{X} an. Die Formel für die Varianz s^2 lautet:

$$s^2 = \frac{\sum_{i=1}^N (X_i - \bar{X})^2}{n - 1} \quad (2)$$

Da die Varianz jedoch andere Einheiten als die Messgrößen verwendet, da der quadratische Abstand angegeben wird, wird häufiger ihre Wurzel, die **Standardabweichung** genutzt. Sei X_i ein Messpunkt aus der Menge aller n Messpunkte, dann gilt für die Standardabweichung:

$$s = \sqrt{\frac{\sum_{i=1}^N (X_i - \bar{X})^2}{n - 1}} \quad (3)$$

Für eindimensionale Mengen von Messpunkten reichen diese Angaben zur Beschreibung aus. Da bei einigen Operationen allerdings die Beziehung zwischen zwei Messmengen und vor allem deren Einfluss aufeinander benötigt wird, gibt es für diese Aussage die **Kovarianz**. Seien X_i Messpunkte aus der Menge $M1$ und Y_i Messpunkte aus der Menge $M2$, dann gilt:

$$c = \frac{\sum_{i=1}^N (X_i - \bar{X})(Y_i - \bar{Y})}{n - 1} \quad (4)$$

Aus dem Ergebnis dieser Formel lässt sich die Abhängigkeit der Messpunkte $M2$ von $M1$ ableiten. Ist das Vorzeichen von c positiv, dann steigt $M2$ wenn $M1$ steigt, bei negativem Vorzeichen sinkt $M2$ wenn $M1$ steigt. Sind zwei Variablen voneinander stochastisch unabhängig, dann ist das Ergebnis 0. Der Umkehrschluss, also die Annahme der stochastischen Unabhängigkeit bei einem Ergebnis von 0 gilt allerdings nicht, da z.B. x^2 und x voneinander abhängig sind, die Gleichung allerdings 0 ergibt.

2.2 Ausgleichsrechnung

Werden Messpunkte auf ein Diagramm aufgetragen, erkennt man leicht Formen, die durch diese Messpunkte gebildet werden, wie z.B. einen linearen Kurvenverlauf in Abb.1. Um die Eigenschaften dieser Gruppierungen zu untersuchen, werden mittels Ausgleichsrechnungen (auch Fittingverfahren genannt) Modellkurven durch die Messpunkte gelegt. Dabei braucht die Parameter enthält, die während der Ausgleichsrechnungen bestimmt werden, als auch einen Fehlerwert (Residuum) ausgibt, der kleiner wird, je mehr das Modell der tatsächlichen Funktion entspricht. Während eine einfache Modellkurve, wie z.B. eine Linie, ein leicht bestimmbares Modell der Form $ax + by + c = 0$ besitzt, können die funktionalen Modelle bei komplizierten Kurven, wie z.B. Ellipsen im 3D Raum, sehr viele Parameter enthalten.

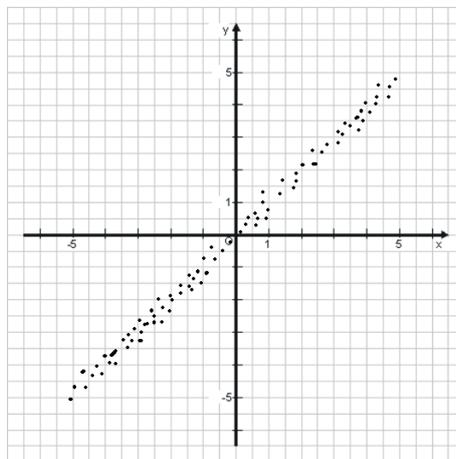


Abbildung 1: Messergebnis, dessen Messpunkte augenscheinlich eine Gerade ergeben und zu deren weiterer Untersuchung eine Ausgleichsgerade angelegt werden kann

Neben dem funktionalen Modell gibt es bei Ausgleichstechniken auch ein stochastisches Modell, das angibt, wie stark einzelne Punkte für die Ausgleichung gewichtet werden. Dies ist vor allem dann hilfreich, wenn man bei der Messung bestimmen kann, wie sicher man sich einiger Aussagen ist. So kann man z.B. beim Fitting einer Kurve an Winkelwerte und Koordinaten die Koordinaten stärker gewichten als die Winkel wenn diese besser gemessen werden können.

Zur Bestimmung der einzelnen Parameter der Modelle gibt es viele unterschiedliche Verfahren. Eines der bekanntesten und ältesten ist die Methode der kleinsten Quadrate von Gauss, die mit dem Abstand der Messpunkte von der Modellkurve arbeitet, den Residuen. Sie gibt an, dass alle Residuen die gleiche Varianz haben, von jedem anderen Residuum stochastisch unabhängig und im Mittel Null sein sollen. Sei $f(\vec{p}, \vec{z})$ die Modellgleichung, bei der die Dimension m des Parametervektors \vec{p} die Anzahl der zu schätzenden Parameter angibt und der Messwertvektor \vec{z} die Messwerte enthält, dann muss somit die folgende Funktion minimiert werden:

$$F = \sum_i^N f^2(\vec{p}, \vec{z})$$

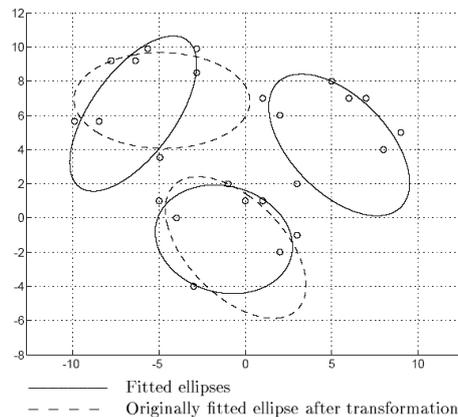


Abbildung 2: Beispiel einer euklidischen Transformation und das dadurch veränderte Fittingergebnis bei algebraischem Entfernungsmaß
Quelle: [GG96]

Die Bestimmung der Residuen kann durch die Messung des algebraischen Abstandes $f(x) = 0$ oder durch die Messung des euklidischen Abstandes durchgeführt werden, welcher die kürzeste Entfernung eines Punktes zur Modellkurve darstellt (siehe auch Abb.3). Die algebraische Form hat den Nachteil einer ungenauen Messung bei vielen Formen. Sind z.B. die Messpunkte in Form einer vertikalen Linie angeordnet, kann der Abstand der Punkte nicht mit der Geradengleichung als algebraischem Abstand berechnet werden. Zudem sind Fittingverfahren mit algebraischem Abstand oft nicht invariant gegenüber euklidischen Transformationen, ergeben also je nach Lage des Objektes ein anderes Ergebnis. Ein Beispiel hierfür findet sich in Abb.2, bei dem die eigentliche Lage der Ellipse gestrichelt dargestellt ist und die durch Drehung der Punkte bei algebraischem Entfernungsmaß falsch erkannte Ellipse als durchgehende Linie gezeichnet wurde. Zudem können Punkte mit unterschiedlicher Positionen im Raum aufgrund des algebraischen Entfernungsmaßes unterschiedliche stark gewichtet werden. Somit kann ein Modell in der Praxis stärker an Punkte angefitet werden, die in Bereichen geringer Krümmung des Modells liegen.

Der Vorteil der algebraischen Entfernung liegt in der einfachen Berechenbarkeit, da keine Abstandsberechnungen durchgeführt werden müssen, sondern das Funktionsergebnis das Gütemaß darstellt. Für viele Grundkörper wie Linien (siehe [For03] Kapitel 15.2) oder Ellipsen (siehe Kapitel 3.2) gibt es direkte Ausgleichsverfahren, die ohne Iterationen auskommen und damit sehr schnell berechenbar sind. Die genaueste Messmethode ist die Berechnung des euklidischen Abstandes, also des direkten Weges eines Punktes zur Modellkurve. Sie hat allerdings bei komplizierten Funktionen, wie z.B. Ellipsen den Nachteil einer sehr aufwendigen und zeitintensiven Berechnung, da iterative Methoden benötigt werden. Sei d_i der euklidische Abstand, dann gilt:

$$F(\vec{p}) = \sum_i^n d_i^2$$

Faber und Fisher beschreiben zwar in [FF02] einen Weg, den euklidischen Abstand von Punkten zu einer Ellipse direkt auszurechnen. Allerdings ist laut selbigem Dokument die Rechenzeit immer noch ca. 5-19x so hoch wie die für den algebraischen Abstand. Je nach Anwendungsfall muss somit zwischen Genauigkeit und schneller Berechenbarkeit abgewogen werden.

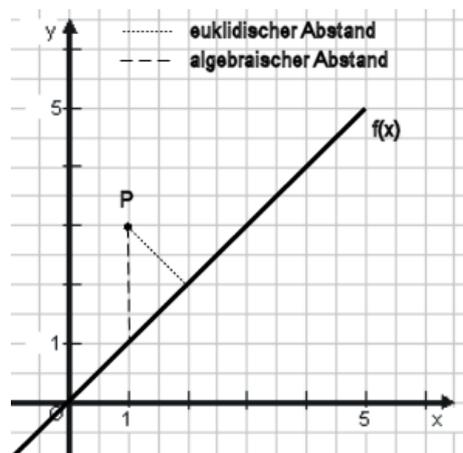


Abbildung 3: Unterschied zwischen euklidischem und algebraischem Abstand am Beispiel einer Gerade, bei der der algebraische Abstand $f(x) - y$ entspricht

Algebraischer Abstand von P zu $f(x)$: 2 Einheiten
 Euklidischer Abstand von P zu $f(x)$: $\sqrt{1}$ Einheiten

2.3 RANSAC

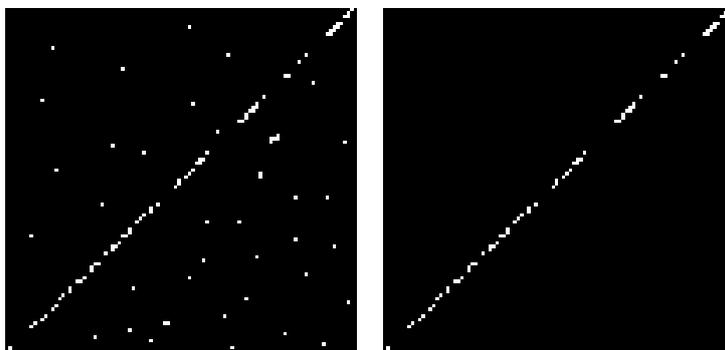


Abbildung 4: Wirkung von RANSAC auf fitting: Links die Messwerte vor RANSAC, rechts die nach dem RANSAC verbliebenen Punkte

Random Sample Consensus (RANSAC) ist ein Fischler und Bolles [FB81] entwickeltes robustes mathematisches Schätzverfahren zur Entfernung von Ausreißern aus einer Beobachtungsmenge. Bevor Modellkurven in einem Bild durch eine Ausgleichsrechnung an Punktwolken angefitzt werden können, sollte entschieden werden, welche der Messpunkte zu der gesuchten Punktwolke gehören und welche als Ausreißer gelten. Abb.5 zeigt den Ablaufplan des RANSAC Verfahrens.

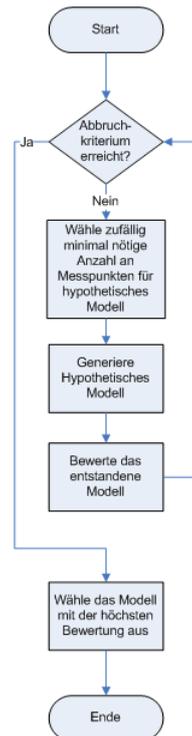


Abbildung 5: Ablaufplan des RANSAC Algorithmusses

Durch die häufige Wiederholung des Verfahrens mit zufälligen Messpunkten kann man mit einer im voraus berechenbaren Wahrscheinlichkeit davon ausgehen, bei mindestens einem der Durchläufe nur gültige Messpunkte zu nutzen, womit dieser die höchste Bewertung erlangt. Um die Wahrscheinlichkeit des Auftretens eines nur aus gültigen Messpunkten bestehenden Durchlaufs zu erhöhen, sollte nur die für die Bildung einer Modellkurve minimal nötige Anzahl von Messpunkten gewählt werden. Die Bewertungsfunktion und das Abbruchkriterium sind je nach Anwendungsfall zu definieren.

Im Beispiel von Abb.4 wird das Verfahren anhand einer Punktmenge dargestellt. Hier wurden 100 mal 2 Punkte ausgewählt und durch diese eine drei Pixel dicke Linie gezogen. Die Bewertungsfunktion bestand darin, die Punkte zu zählen, die von der Linie getroffen wurden. Nach dem Ablauf der 100 Durchläufe verblieben nur die Punkte im Bild, die von der Linie mit der höchsten Bewertung berührt wurden. Alle weiteren Punkte wurden als Ausreißer aus dem Bild entfernt.

2.4 Houghtransformation

Die 1962 von P.V.C. Hough entwickelte Houghtransformation ([Hou62]) ist ein häufig verwendetes Verfahren zur Suche bekannter Formen in Bildern mittels ihrer Parameterdarstellung. Dafür werden für jeden gefundenen Punkt im Bild alle möglichen Parameterkombinationen der gesuchten Funktion in einem Akkumulatorraum gespeichert, die diesen Pixel durchlaufen. Da die tatsächlich gesuchte Kombination dabei am stärksten akkumuliert wird, kann diese danach durch die Suche nach dem höchsten Wert des Parameterraumes gesucht werden. Veranschaulicht wird dies durch das folgende Beispiel der Suche nach Geraden im Bild.

Das Schema des Verfahrens wird hier an einer Liniensuche in Abb.6 dargestellt. Linien können in ihrer Hesse'schen Normalform als Gleichung $x \cos \phi + y \sin \phi = d$ dargestellt werden. Dabei stellen ϕ und d die Parameter der Linien dar, aus denen der Parameterraum (ϕ, d) aufgebaut wird. Nun wird das Bild durchsucht und für jeden gefundenen Pixel im Bild alle Parameterkombinationen im (ϕ, d) Raum akkumuliert, die eine Linie durch diesen Punkt bilden. Existiert eine bestimmte Gerade im Bild, wird deren Parameterraum öfter als alle anderen Parameter akkumuliert und sie kann somit aus dem Parameterraum durch die Suche nach dem höchsten Eintrag gefunden werden. In Abb.6 sind im Linken Bild das Originalbild mit den darauf liegenden Linien dargestellt, deren Entsprechung rechts im Parameterraum dargestellt sind.

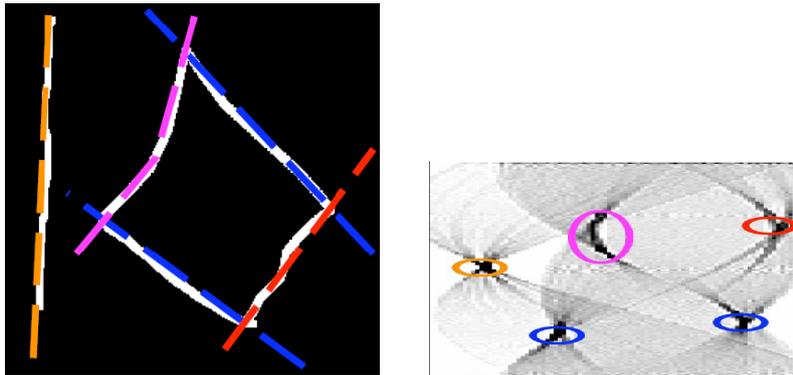


Abbildung 6: Linien im Bild und ihre Entsprechung im Parameterraum

Quelle: [Fre05]

Das größte Problem bei diesem Verfahren ist der enorme Speicherverbrauch durch den Parameterraum. So sind Linien mit 2 Parametern (ϕ, d) schnell bestimmbar, während Kreise schon eines dreidimensionalen Raumes bedürfen (Mittelpunkt (x, y) , Radius).

2.5 Grundlegende Operationen der Bildverarbeitung

Die folgend vorgestellten Bildverarbeitungsalgorithmen sind grundlegende Operatoren, die in vielen Literaturquellen (z.B. [Jäh02], [Dav96]) und Vorlesungsskripten (z.B. [Jia02]) gefunden werden können. Daher werden hier nur die Prinzipien zum Verständnis vorgestellt und nicht die Implementierungen der Algorithmen.

2.5.1 Schwellwertverfahren

Viele Verfahren benötigen die Separierung eines Bildes in zu bearbeitende Teile und den nicht zu bearbeitenden Hintergrund. Diese Teilung kann durch Schwellwerte realisiert werden, die die Grenze zwischen den Zuständen darstellen. Die Schwierigkeit besteht darin, diesen Wert so zu setzen, dass alle gewünschten Elemente auf der einen Seite und die nicht gewünschten Elemente auf der anderen Seite der Grenze stehen. Schon ein Graustufenbild stellt ein Problem bei der Setzung des richtigen Wertes dar, da ein einmal festgelegter Schwellwert schon durch eine einfache Helligkeitsänderung des Bildes falsche Ergebnisse liefert. Somit braucht man für diese Bilder helligkeitsinvariante automatische Schwellwertverfahren.

Ein mögliches Verfahren hierfür ist die automatische Schwellwernerkenntung nach

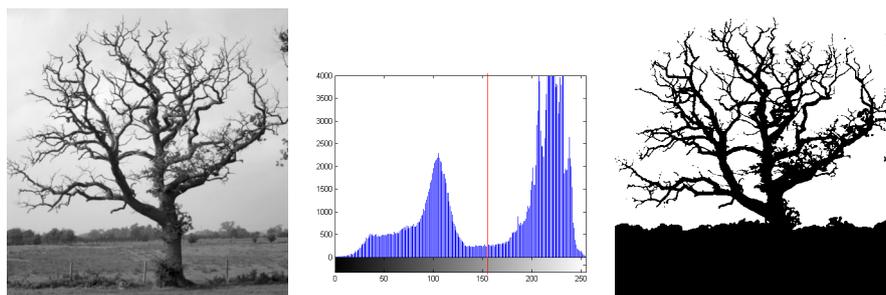


Abbildung 7: Beispielbild für Otsu: Links ein Graustufenbild eines Baumes in der Landschaft, in der Mitte das dazugehörige Histogramm mit der Einzeichnung des Wertes, den Otsu als Schwellwert ermittelt hat, rechts das Ergebnisbild mit schwarz als Werte unterhalb des Schwellwertes und weiss als Werte oberhalb des Schwellwertes.

Quelle: <http://de.wikipedia.org/wiki/Bild:DeadTree.jpg>

Otsu [Ots79], die invariant unter linearer Helligkeitstransformation ist und somit die oben gestellte Bedingung erfüllt. Sie eignet sich sehr gut, um Bilder in zwei Bereiche einzuteilen, wie es für ein schwarz-weißes Kantenbild erforderlich ist. Die Schwierigkeit hierin besteht in einer Einschätzung, wie die Helligkeitsbereiche getrennt werden sollten. Otsu verwendet dafür einen statistischen Ansatz, bei dem versucht wird, die Abweichung der Mittelwerte der beiden Ergebnisklassen K_1 und K_2 im Vergleich zum Mittelwert aller Grauwerte möglichst groß zu halten, während die Varianzen der Klassen möglichst gering sein sollen. In Abb.7 ist dies gut im Histogramm zu erkennen, in dem der Mittelwert markiert ist, welcher die am häufigsten Auftretenden Pixel gut sichtbar voneinander trennt.

Problematisch ist das Verfahren bei Objekten vor einem Farbverlauf wie in Abb.8. Durch diesen sind im Histogramm sehr viele Pixel gleichmäßig verteilt, so dass die

relativ dazu wenigen Pixel des Gesichts die Gleichverteilung nicht mehr sehr stark beeinflussen können. Das Verfahren nach Otsu setzt daher den Schwellwert auf die Mitte des Farbverlaufes, wodurch die eigentlich vom Hintergrund zu separierende Form, das Gesicht, zerstört wurde. In der Praxis wirkt sich dies allerdings nicht sehr stark aus, da solche Bilder nur sehr selten vorkommen.



Abbildung 8: links: Bild vor Otsu rechts: Ergebnis nach Otsu

Bildquelle: <http://de.wikipedia.org/wiki/Schwelldwertverfahren>

2.5.2 Kontrastverstärkung

Durch schlechte Belichtung oder nahezu gleichfarbene Gegenstände in einem Bild kommt es häufig vor, dass sich die einzelnen Bildelemente farblich nicht sehr gut voneinander unterscheiden, was vor allem bei Schwellwertverfahren zu Problemen führen kann. Somit muss eine Kontrastverbesserung durchgeführt werden, um die Grauwertunterschiede eines Bildes stärker hervorzuheben.

Sei g_{xy} der Grauwert des Pixel an der Stelle (x/y) , c die gewünschte Kontraständerung und l die gewünschte Helligkeitsänderung, dann erreicht man eine einfache lineare Kontraständerung mit der folgenden Formel:

$$g'_{xy} = (g_{xy} * c) + l$$

Da man bei dieser Methode feste Werte verwendet werden, ist sie jedoch nicht automatisiert anwendbar. Somit würde zwar mit einer Kontrastverstärkung der Kontrast bei einem dunklen Bild verbessert, bei einem hellen Bild allerdings im schlimmsten Fall mit derselben Operation alle Pixel auf den Maximalwert von 255 gesetzt werden. Hier bietet sich stattdessen eine Histogrammebnung an. Bei dieser werden alle Grauwerte so skaliert, dass sie mit der gleichen Häufigkeit im Histogramm erscheinen. Sind von einem Grauwert hierfür zu wenig Pixel vorhanden, werden andere Pixel, die entweder zufällig, oder in Abhängigkeit von der Nachbarschaft zum bearbeiteten Pixel, ausgewählt werden können, diesem Grauwert zugeordnet. Die Grauwerte werden somit gleichmäßig über das Bild verteilt, allerdings können im Histogramm an einigen Grauwerten Lücken entstehen. Somit ergibt sich ein Ergebnis, wie in Abb.9 zu erkennen.

Da die Histogrammebnung immer alle Punkte im Bild berücksichtigt, ist sie sehr anfällig für gleichfarbene Hintergründe, die das Originalhistogramm dominieren

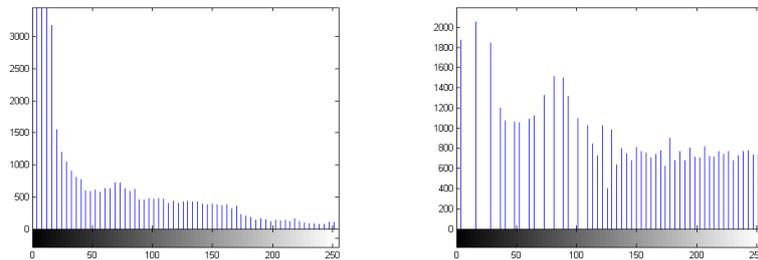


Abbildung 9: links: Histogramm eines Bildes, rechts: Histogramm desselben Bildes nach einer Histogrammgebung

und das geebnete Histogramm somit in ihren Bereich hineinziehen.

Um diesen Nachteil auszugleichen gibt es die Adaptive Histogramm Equalization (AHE) nach [PAA+87], die das Bild in einzelne Kacheln unterteilt. Die Kachelgröße kann hierbei beliebig festgelegt werden, wobei eine Kachelgröße gleich der Bildgröße eine normale Histogrammgebung darstellt. Die Histogramme werden bei der AHE innerhalb der Kacheln geebnet, wodurch auf lokale Begebenheiten sehr viel besser eingegangen werden kann als bei globalen Verfahren und der Hintergrund nicht mehr das Histogramm dominiert. Die entstandenen Grauwerte sind jedoch nur Zwischenwerte, da bei einer alleinigen Anwendung dieses Verfahrens die Ränder der Kacheln eindeutig zu erkennen wären. Zur Schaffung von weichen, nicht sichtbaren Übergängen werden daher die Grauwerte in den Kacheln abhängig von ihrer Entfernung zur Mitte der Nachbarkacheln bilinear interpoliert. Obwohl der Kontrast der Ergebnisbilder bei AHE sehr viel besser als bei der Histogrammgebung ist, hat dieses Verfahren für die Kantenerkennung den großen Nachteil, dass das Hintergrundrauschen in unifarbenen Bildelementen extrem verstärkt wird.

Diesen Nachteil gleicht die Contrast Limited Adaptive Histogramm Equalization



Abbildung 10: links: Eingabebild rechts: Nach CLAHE

(CLAHE) [Zui94] aus, eine Erweiterung von AHE, die die maximale Kontraständerung der Pixel auf eine Einstellbare Größe festlegt. Somit werden zu große Sprünge in den Grauwerten verhindert und das Bildrauschen wird in unifarbenen Bereichen nicht unnötig verstärkt. Ein Beispiel für die CLAHE ist Abb.10, bei dem ohne CLAHE der Reifen nicht vom Hintergrund zu unterscheiden ist und sich nach der

Anwendung deutlich vom Hintergrund abhebt.

2.5.3 Bildglättung

Bildglättung wird dort eingesetzt, wo Rauschen das Bild stört und die Kantenerkennung zu viele falsche Kanten erkennen lässt. Für die Bildglättung gibt es Faltungsoperatoren, wie z.B. Average- oder Gaussfilter, oder Rangordnungsfiler, wie dem Medianfilter. Zur Darstellung der unterschiedlichen Funktionsweise dieser beiden Filterarten werden der Averagefilter und der Medianfilter hier exemplarisch erklärt.

Faltungsoperatoren benutzen eine Faltungsmatrix, mit der der Einfluss der benachbarten Bildpunkte beschrieben wird und einen Faktor, der das Ergebnis auf den Wertebereich von 0-255 normalisiert. Bei dem Averagefilter sieht dies wie folgt aus:

$$g(x) = \frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Jeder Wert der Nachbarpixel fließt also mit der gleichen Gewichtung in den neuen Pixelwert ein.

Im Gegensatz zu den Faltungsoperatoren nutzen Rangordnungsfiler wie der Medianfilter die sortierten Grauwerte der benachbarten Pixel für die Grauwertbestimmung. Der Medianfilter wählt aus den sortierten Werten immer den mittleren Grauwert. Da kleine Störungen, wie beim Bildrauschen sich meist nur auf ein einziges Pixel auswirken und somit bei den sortierten Werten nicht den mittleren Wert darstellen, eignet sich dieser Filter daher gut, um ein Bildrauschen zu unterdrücken.

Ein bildlicher Vergleich findet sich in Abb.11.



Abbildung 11: von links nach rechts: Ausgangsbild, Ergebnis mit Averagefilter, Ergebnis mit Medianfilter

Quelle: Matlab 7.1

2.5.4 Kantendetektion

Verfahren zur Kantendetektion dienen der Suche sowohl nach Kanten in einem Bild, als auch nach deren Ausrichtung. Der bekannteste Kantendetektor ist der Sobeloperator. Sei S_x der Sobeloperator in X-Richtung und S_y der Sobeloperator in Y-Richtung, dann wird das Bild mit dem folgenden Operator gefaltet:

$$S_x = \frac{1}{8} \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \quad S_y = \frac{1}{8} \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \quad (5)$$

Sei nun gS_x die Kantenstärke eines Pixels im Kantenbild des S_x -Operators und gS_y die Kantenstärke eines Pixels im Kantenbild des S_y -Operators dann wird der Gradient ϕ wie folgt berechnet:

$$\phi = \arctan\left(\frac{gS_y}{gS_x}\right)$$

Ein sehr häufig genutzter Kantenfilter ist auch der Cannyfilter [Can86], der vor dem Sobeloperator noch einen gaußschen Weichzeichner einsetzt, durch eine Non-Maximum-Suppression die Kantendicke auf 1 reduziert und den Schwellwert durch Hysteresis bestimmt. Hysteresis ist hier ein Verfahren, bei dem 2 Schwellwerte eingesetzt werden. Alle Punkte mit höheren Werten als der 2. Schwellwert werden als eindeutige Kantenpunkte gewertet, während die Werte zwischen dem ersten und zweiten Schwellwert nur dann einen Kantenpunkt darstellen, wenn ein Nachbar ein eindeutiger Kantenpunkt ist.

Beide Filter besitzen laut Jähne (siehe [Jäh02] Kapitel 13.3.6) einen Gradientenfehler von ca. 7° . Zur Erhöhung der Gradientengenauigkeit definiert Jaehne einen optimierten Sobeloperator (siehe [Jäh02] Kapitel 12.5.5) mit der folgenden Faltungsmatrix, der einen Gradientenfehler von nur noch 0.5° besitzt. Sei OS_x der optimierte Sobeloperator in X-Richtung und OS_y der optimierte Sobeloperator in Y-Richtung, dann gilt:

$$S_x = \frac{1}{32} \begin{pmatrix} 3 & 0 & -3 \\ 10 & 0 & -10 \\ 3 & 0 & -3 \end{pmatrix} \quad S_y = \frac{1}{32} \begin{pmatrix} 3 & 10 & 3 \\ 0 & 0 & 0 \\ -3 & -10 & -3 \end{pmatrix} \quad (6)$$

2.5.5 Eckendetektion

Da die Suche nach Ecken ein wichtiges Gebiet der Bildverarbeitung ist, vor allem in der Merkmalsextraktion, gibt es hierfür viele Verfahren. Ein bekannter Eckendetektor ist der Harrisdetektor [HS88]. Dieser nimmt für jeden Pixel des Bildes die Umgebung und beobachtet die Intensitätsänderung des Ausschnittes bei kleinen Drehungen. Da die größte Änderung bei einer Ecke stattfindet, können die Änderungen in einen Parameterraum gespeichert und hier mittels eines Schwellwertes die Ecken bestimmt werden. Dieses Verfahren eignet sich zwar gut für die Merkmalsextraktion eines Bildes, liefert allerdings im Sinne einer reinen Eckensuche zu viele Fehlerkennungen, wie in Abb.12 zu erkennen.

In einem Vergleich von [Jia05] werden Algorithmen wie SUSAN ([SB97]) und Kovesi ([Kov02]) gezeigt. Da diese jedoch wegen ihren globalen Schwellwerten alle an die gegebenen Bilder angepasst werden müssen, eignet sich für eine automatisierte Vorgehensweise der Detektor von He/Yung ([HY04]) besser. Dieser umgeht

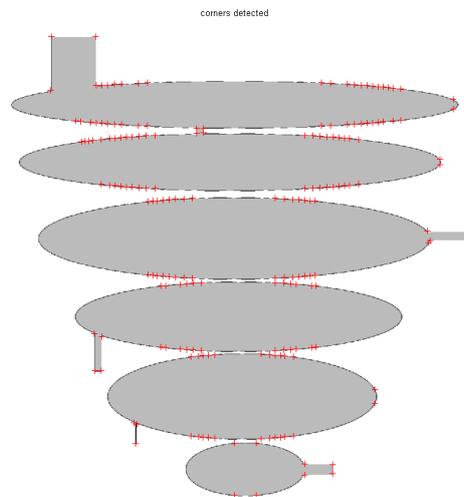


Abbildung 12: Ergebnis des Harris Kantendetektors mit einem Schwellwert von 10000. Die gefundenene Ecken sind rot markiert

das Problem durch adaptive lokale Schwellwerte und zeigt in Tests ein sehr gutes Ergebnis (siehe auch Abb.13). Bei ihrem Ansatz wird das Kantenbild in ein Konturbild umgewandelt und in diesem an jeder T-Kreuzung der Konturen die Krümmung berechnet. Bei allen Stellen, an denen die Krümmung mindestens doppelt so groß wie diejenige der lokalen Nachbarschaft ist, liegt eine Ecke vor.

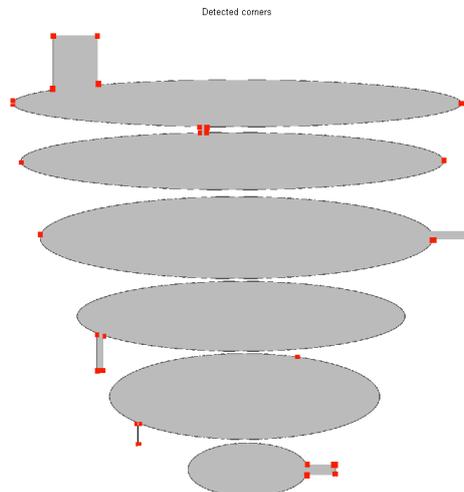


Abbildung 13: Mit He/Yung's Eckendetektor gefundene Ecken (Zur besseren Erkennung sind die detektierten Ecken mit rot nachgezeichnet worden)

2.5.6 Labeling

Labeling dient der Zusammenfassung benachbarter Pixel mit ähnlichen Eigenschaften wie z.B. gleicher Farbe, zu Objekten.

Ein einfaches Labelingverfahren ist der Floodfill Algorithmus. Bei diesem Verfahren wird das Bild mit dem Algorithmus aus Abb.14 durchlaufen.

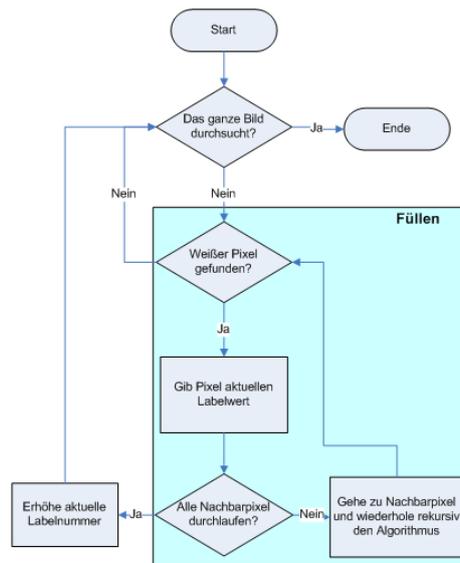


Abbildung 14: Ablaufplan des Grassfire Algorithmusses zum Labeln

Das Verfahren ist zwar sehr einfach zu implementieren, durch den Rekursionsoverhead aber sehr langsam. Ein schnelleres Verfahren hingegen stellt der Union-Find Algorithmus dar. Bei diesem wird das Bild erst zeilenweise durchlaufen, wobei alle zusammenhängenden Pixel einer Zeile als gemeinsames Intervall gekennzeichnet werden. In einem zweiten Durchlauf werden dann alle sich berührenden Intervalle miteinander zu einem gemeinsamen Label verbunden. Das Ergebnis findet sich in Abb.15, bei dem jedem Label eine eigene Farbe zugewiesen wurde.

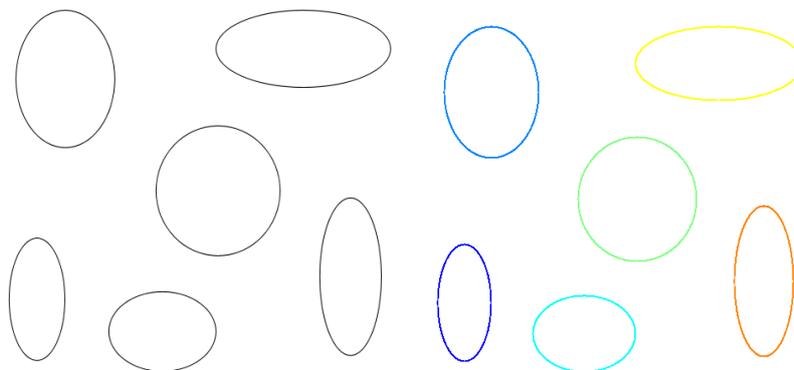


Abbildung 15: links: ungelabelt rechts: gelabelt

2.5.7 Erosion, Dilatation Opening und Closing

Erosion und Dilatation sind Verfahren auf Binärbildern, die die Nachbarschaften eines Pixels in Abhängigkeit von einem Strukturelement verändern.

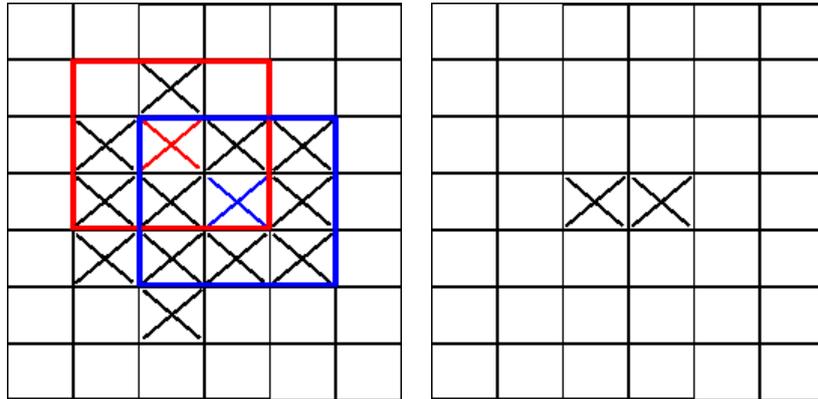


Abbildung 16: Beispiel einer Erosion mit einem quadratischen Strukturelement. Der rote Pixel verfügt über zu wenig Nachbarn und wird aus dem Bild gelöscht, der blaue Pixel besitzt genügend Nachbarn und verbleibt im Bild. Das rechte Bild zeigt das Ergebnis für alle Pixel

Erosion

Erosion ist ein Verfahren zum Verkleinern von Strukturen, bei dem Pixel im Bild abhängig von ihrer Nachbarschaft entfernt werden. Dazu wird für jeden Pixel im Bild anhand eines Strukturelementes geprüft, ob er genügend viele Nachbarpixel besitzt. Füllt die Anzahl und Anordnung der Nachbarpixel das Strukturelement aus, so verbleibt der Pixel im Bild, ansonsten wird er entfernt. Veranschaulicht wird dies in Abb.16.

Dilatation

Bei der Dilatation werden alle Nachbarn jedes Pixels, die das Strukturelement abdeckt, gesetzt. Ein Beispiel findet sich in Abb.17

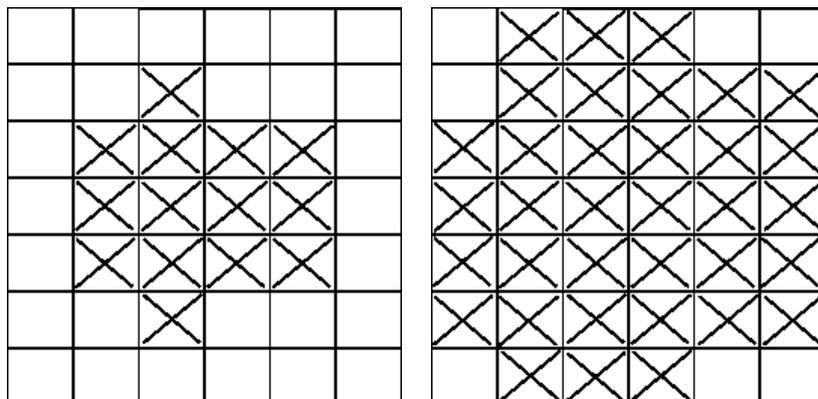


Abbildung 17: Beispiel einer Dilatation. Das Ausgangsbild ist links, das Ergebnisbild rechts zu sehen

Opening

Beim Opening wird erst eine Erosion auf das Bild angewandt, danach eine Dilatation. So können Strukturen gelöscht werden, die kleiner als das Strukturelement sind, um z.B. Rauschen aus einem Bild zu entfernen.

Closing

Beim Closingoperator wird zuerst eine Dilatation auf das Bild angewandt, danach eine Erosion. So können Strukturen miteinander verbunden werden, die um die Größe der Dilatation voneinander entfernt sind. Praktisch ist dies z.B. wenn Kanten durch Schwellwerte an einigen Stellen Lücken aufweisen.

2.6 Messstand

Der für die Aufnahmen an den Testrädern und -Autos genutzte Messstand besteht aus einem eigens konstruierten Gerüst mit 2 CCD-Kameras und einem Beamer für verschiedene Beleuchtungsmuster und -stärken. Der Aufbau ist in Abb.18 zu erkennen.



Abbildung 18: Der verwendete Messstand

Die Kameras verwendeten CCD Chips zur Digitalisierung der aufgenommenen Bilder, die mit einer Auflösung von 1 Megapixel als Graustufenbild im PNG Format gespeichert werden.

2.6.1 Aufbau einer CCD Kamera

Eine CCD Kamera verwendet zeilen- oder matrixförmig angebrachte Photosensoren, auf die das auftreffende Licht fällt. Die Photonen des Lichtes werden in einer Halbleiterschicht absorbiert und erzeugen dabei negative Ladungsträger, die an einer positiv geladenen Elektrode akkumuliert werden. Die Ladung der Elektrode wird nun wie in einer Eimerkette entlang der Elektroden weiter gereicht (dieses Vorgehen wird auch als Eimerkettenprinzip oder CCD-Prinzip (charged coupled device) bezeichnet, bis sie auf ein Ausleseregister trifft, wo sie weitergeleitet wird, um zu einem elektrischen Spannungssignal umgewandelt zu werden. Dies wird von einem Potentiometer ausgelesen und quantisiert als digitale Information ausgegeben.

Ein Effekt der bei CCD-Sensoren auftreten kann ist das so genannte Blooming. Dabei wird in den Sensoren mehr Ladung erzeugt, als in den Elektroden gespeichert werden kann und die überschüssige Ladung tritt in die umliegenden Sensoren über. Ein Beispiel hierfür findet sich in Abb.19 an der hellen Felge.

In Consumerkameras taucht Blooming mittlerweile nicht mehr auf, da hier die überschüssige Ladung abgeleitet wird. Da hierbei aber schon Ladung vor der Überfüllung abgeleitet werden kann, gibt die Ladung nicht mehr die korrekte Lichtmenge wieder, wodurch sich das Verfahren nicht für genaue Messungen eignet.

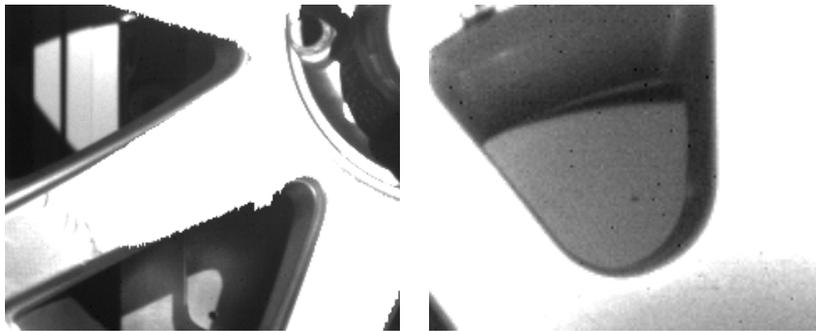


Abbildung 19: Blooming

Abbildung 20: Kamerarauschen

Bei den verwendeten digitalen Kameras kann man die Ausgabe des Bildes an zwei Stellen beeinflussen, der Shutterzeit und dem Gain. Die Shutterzeit gibt die Zeit an, während der das Licht in die Kamera gelassen wird. Eine höhere Shutterzeit erhellt das Bild somit, allerdings zum Nachteil eines starken Risikos des Auftretens von Blooming. Der Gain ist das Maß für die innere Verstärkung der Kamera. Ein höherer Gain sorgt für ein helleres Bild ohne das Auftreten von Blooming, allerdings wird zusammen mit dem Bild auch das Kamerarauschen verstärkt, was die weitere Bearbeitung des Bildes erschweren kann. Das Rauschen ist in Abb.20 als im Bild verstreute schwarze Pixel zu erkennen.

2.6.2 Stereoskopie

Verfahren zur Stereorekonstruktion nutzen die Daten zweier Kameras, um aus deren Bildern die Tiefeninformation aller Punkte zu berechnen, die in beiden Bildern auftreten. Veranschaulicht wird dies in Abb.21 bei der Stereonormalfall dargestellt wird, bei dem beide Kameras parallel zueinander auf gleicher Höhe aufgestellt werden. Somit kann die Tiefeninformation durch Messung der x -Parallaxe px' bestimmt werden kann. Dabei gilt für die Tiefeninformation Z eines Punktes P die Formel:

$$Z = h = \frac{b \cdot c}{x' - x''} = \frac{b \cdot c}{px'} \quad (7)$$

Der schwierigste Teil des Verfahrens ist die Korrespondenzsuche, also das Bestimmen zweier Punkte aus den jeweiligen Bildern, die denselben Punkt des 3D Raumes darstellen. Hierzu gibt es verschiedene Verfahren, die diese Arbeit jedoch nicht betreffen und daher nicht weiter erläutert werden. Das einzige dieser Verfahren das die hier vorgestellte Arbeit betrifft, da es sich auf die Eingangsbilder auswirkt, ist die Nutzung von Strukturbeleuchtung, bei der Lichtmuster so auf die zu erkennende Fläche projiziert werden, dass man anhand der somit bekannten Kantenbildung die Punkte der Bilder besser einander zuordnen kann.

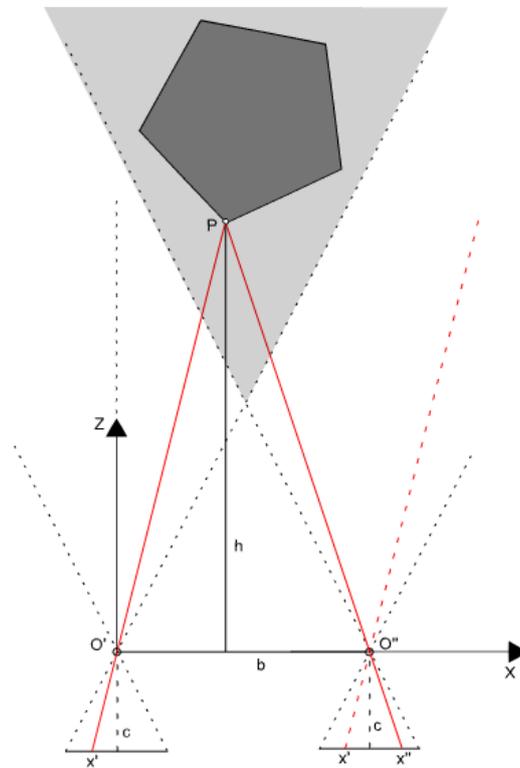


Abbildung 21: schematische Darstellung des Stereonormalfalls

Quelle: eigene Nachbildung einer Darstellung aus [Luh00] Seite 310

3 Verfahren zur Ellipsendetektion

Da zur Beschreibung einer Ellipse 5 Parameter gebraucht werden (z.B. Schwerpunkt x und y , Hauptachse, Nebenachse, Winkel der Achsen), ist die effiziente Suche nach Ellipsen in einem Bild durch diesen fünfdimensionalen Suchraum immer noch ein aktuelles Problem der Bildverarbeitung. Eine Reihe von Verfahren sind zur Lösung dieses Problems entwickelt worden, die in drei Bereiche eingeteilt werden können.

- Ellipsendetektion nach Hough
Diese Verfahren nutzen die Houghtransformation, bilden also die Parameter in einem Parameterraum ab und wählen die am stärksten akkumulierte Parameterkombination. Der Vorteil dieser Verfahren ist die Robustheit gegenüber Ausreißern und die Fähigkeit, mehr als eine Ellipse im Bild zu erkennen. Der Nachteil ist die lange Berechnungszeit und die schwache Genauigkeit.
- Fittingverfahren
Für die Ellipsensuche gibt es viele iterative und einige direkte Fittingverfahren. Der Vorteil ist die Genauigkeit und die, bei den direkten Verfahren, hohe Geschwindigkeit der Erkennung. Der Nachteil der Verfahren ist die schwache Robustheit gegen Ausreißer und die Beschränkung des Ergebnisses auf eine einzige Ellipse. Im Gegensatz zur Houghtransformation muss für ein Fittingverfahren allerdings schon vor der Anwendung des Verfahrens bekannt sein, welche Punkte zur Ellipse gehören.
- Sonstige Verfahren
Auf andere Verfahren, wie z.B. die Kawaguchi und Nagata genutzten genetischen Algorithmen [KN98], der konturenbasierte Ansatz von Cai et al. [CYW04], oder Yu und Wing, die eine Ellipse als Summe von 4 Kreisteilen darstellen [Yu95] wird hier nicht näher eingegangen, da sie entweder von der Erkennungsleistung oder von der Ausführungsgeschwindigkeit her schlechter als die oben erwähnte Verfahren sind.

Die Verfahren mit Houghtransformation, sowie die Fittingverfahren werden gleichermaßen in der Literatur, sowie in Veröffentlichungen (siehe z.B. [Jia02]) vorgestellt und im nächsten Abschnitt näher erläutert.

3.1 Ellipsendetektion nach Hough

Da für die Beschreibung einer Ellipse in der Ebene mindestens 5 Parameter benötigt werden ist der Parameterraum zu groß um effizient durchsucht werden zu können. Eine Lösung liegt hier im Aufteilen der Parameter, so dass zuerst der Mittelpunkt berechnet wird und danach Haupt- und Nebenachse, sowie der Winkel.

Dazu werden, wie in Abb.22 dargestellt, an 3 Punkten P_1 , P_2 und P_3 Tangenten angelegt. Der Schnittpunkt der beiden Geraden, die nun durch die Schnittpunkte der Tangenten (T_1 und T_2), sowie durch die Mittelpunkte der Punkte P_1 und P_2 , bzw. P_2 und P_3 laufen, bildet das Ellipsenzentrum C .

Zur Berechnung der Haupt- und Nebenachse wird die Ellipse nun in den Ursprung transformiert, um die allgemeine Ellipsengleichung $a(x-p)^2 + 2b(x-p)(y-q) + c(y-q)^2 = 1$ auf die Form $ax^2 + 2bxy + cy^2 = 1$

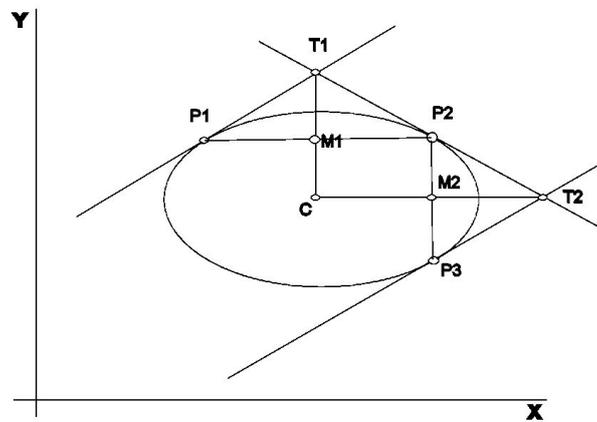


Abbildung 22: Bestimmung des Ellipsenzentrums aus den Tangenten der 3 Punkte P1, P2 und P3. Der Schnittpunkt der Geraden durch die Tangenztangenten T1 und T2, sowie der Mittelpunkte der Punkte M1 und M2 ergibt das Ellipsenzentrum C

zu reduzieren. Die 3 Unbekannte dieser Form können durch Einsetzen der 3 Punkte P1, P2 und P3 in x und y durch das daraus entstehende Gleichungssystem (8) berechnet werden.

$$\begin{aligned} ap_1x^2 + 2bp_1xp_1y + cp_1y^2 &= 1 \\ ap_2x^2 + 2bp_2xp_2y + cp_2y^2 &= 1 \\ ap_3x^2 + 2bp_3xp_3y + cp_3y^2 &= 1 \end{aligned} \quad (8)$$

Damit die dabei entstehende Gleichung auch eine Ellipse und nicht nur den allgemeinen Kegelschnitt darstellt, muss sie Bedingung (9) erfüllen.

$$ac - b^2 = h > 0 \quad (9)$$

Bei $h > 0$ wurde eine Ellipse oder ein Kreis gefunden, $h = 0$ ist eine Parabel und $h < 0$ bedeutet eine Hyperbel (siehe Abb.23).

Der Winkel der Ellipse kann nun durch Einsetzen der bisher gefundenen 4 Parameter in die folgende Formel errechnet werden.

$$\frac{((y - q) * \sin \theta + (x - p) * \cos \theta)^2}{r_1^2} + \frac{((y - q) * \cos \theta - (x - p) * \sin \theta)^2}{r_2^2}$$

Gefundene Ellipsen werden in einem Akkumulator aufgezählt, um so die besten Ellipsen nach dem Suchvorgang bestimmen zu können.

Eine weitere Verbesserung ist die im Skript von Jiang [Jia02] beschriebene Probabilistic Hough Transformation, bei der nur ein kleiner Teil (ca. 2-15%) der gefundenen Pixel analysiert wird.

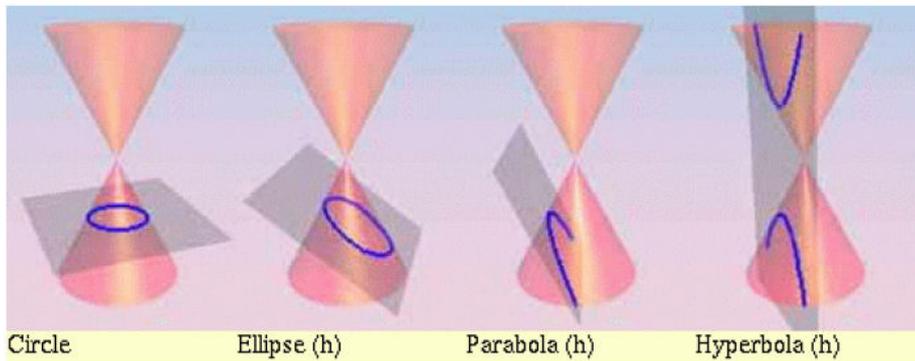


Abbildung 23: Darstellungen im Kegelschnitt
Quelle: [Inv02]

Xu hat in [XOK90] einige Probleme dieser Varianten der Houghtransformation aufgezeigt, zu deren Lösung er die „Randomized Hough Transformation (RHT)“ vorschlägt, die sich seiner Meinung nach allerdings nicht für Ellipsendetektion eignet. Dies wurde jedoch von McLaughlin in [MR98] widerlegt.

Bei der RHT werden die 3 Punkte zufällig gesucht und nach dem Finden einer Ellipse aus der Suchmenge entfernt. Die Suchlänge und damit auch die Wahrscheinlichkeit einer erfolgreichen Suche kann man hier variieren. Sie ist unterteilt in Epochen und Iterationen. Bei jeder Epoche werden die Zufallszahlen neu gebildet und diese dann n Iterationen oft durchlaufen. Eine Implementation der RHT nach McLaughlin mit ausführlicher Beschreibung des Verfahrens findet sich in [Inv02].

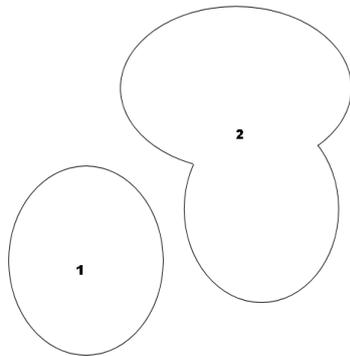


Abbildung 24: gelabelte Ellipsen

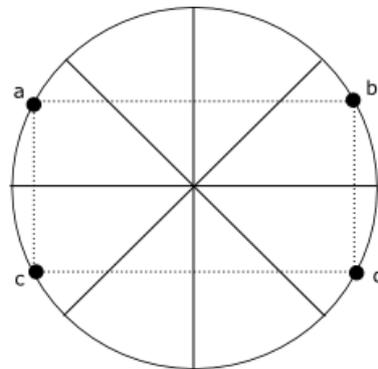


Abbildung 25: Ellipsenregionen

Die zufallsbasierte Suche führt allerdings gerade bei Bildern mit Rauschen zu einer extrem langen Suchdauer von einigen Stunden für ein einziges Bild, da jede Kante berücksichtigt wird. Daher verwenden andere Verfahren wie „Restricted Randomized Hough Transform (RRHT)“ [CL04] Ansätze, die Punkte nicht zufällig, sondern nach einem gewissen Schema auszuwählen.

Die RRHT betrachtet die Ellipsen nach der Kantendetektion als Graphen (siehe Abb.24). Zusammenhängende Linien bekommen ein gemeinsames Label, anhand

dessen sie identifiziert werden. Für die Suche nach Ellipsen kommen somit nur Punkte eines gemeinsamen Labels in Frage, um die Wahrscheinlichkeit zu erhöhen, 3 Punkte einer gemeinsamen Ellipse für eine Suche zu verwenden. Die Geschwindigkeit lässt sich mit dieser Methode laut [CL04] auf durchschnittlich 1/5 der Ausführungsdauer der RHT reduzieren, wobei einfache Störungen durch Rauschen durch das Setzen einer Mindestgröße der Label verhindert werden können. Der Ansatz von Zhang und Liu [ZL03] besteht darin, mit Hilfe der Konvexität der Ellipsenform die Punkte so auszuwählen, dass zur Berechnung Punkte herangezogen werden, die möglichst mit dem zuerst gewählten Punkt auf der horizontalen, oder vertikalen Achse liegen. Im Beispiel von Abb.25 wäre dies bei zuerst gefundenem Punkt a dann die Punkte b, bzw. c.

3.2 Fittingverfahren

Ein einfaches Fitting von Ellipsen ist eine Ausgleichsrechnung nach Kapitel 2.2 bei dem das funktionale Modell eine Ellipsengleichung in Form eines allgemeinen Kegelschnittes darstellt, mit verschiedenen Annahmen, die dafür sorgen, dass das Ergebnis eine Ellipse darstellt und nicht nur einen allgemeinen Kegelschnitt. Es gelten die Ellipsengleichungen:

$$f(p, u) = ax^2 + 2bxy + cy^2 + dx + ey + f = 0 \quad (10)$$

$$p = (x, y) \quad (11)$$

$$u = (a, b, c, d, e, f) \quad (12)$$

Da die euklidische Distanz eines Punktes p_i zur Kurve $f(p, u)$ nur unter hohem Rechenaufwand (siehe Kapitel 2.2) zu bestimmen ist, wird in den meisten Verfahren der algebraische Abstand $e_a(p_i, f(p, u)) = |f(p_i, u)|$ verwendet. Daraus ergibt sich das folgende Optimierungsproblem¹ (siehe auch [FF95] und Kapitel 5.4.4 in [Jia02])

$$\min_u \sum_{i=1}^n [e_a(p_i, f(p, u))]^2 = \min_u \sum_{i=1}^n [f(p_i, u)]^2 \quad (13)$$

Da das Optimierungsproblem in diesem Fall für den allgemeinen Kegelschnitt gilt, müssen für ein Ellipsenfitting Beschränkungen definiert werden. Diese Beschränkungen sind auch das Hauptunterscheidungsmerkmal der unterschiedlichen Ellipsenfittingverfahren. In einem Vergleich verschiedener Verfahren wie Bookstein [Boo79] oder Taubin [Tau91] in [FF95] wird jedoch erwähnt, dass zwar alle diese Algorithmen bei perfekten Messpunkten sehr gute Ergebnisse erzielen, aber sich in der Robustheit gegenüber Rauschen unterscheiden. Zudem erzeugen viele dieser Algorithmen nicht immer Ellipsen.

Eine Lösung erfolgt durch das direkte Ellipsenfitting nach [PFF96]. Da $f(p, u) = 0$ genau dann eine Ellipse ist, wenn $b^2 - 4ac < 0$ gilt, kann folgende Annahme getroffen

¹Da in den Artikeln und in([Jia02]) unterschiedliche Notationsweisen der Formeln auftreten, werden diese hier zu besseren Lesbarkeit einheitlich in der Weise von Jiang aufgeschrieben

werden.

$$b^2 - 4ac = u \underbrace{\begin{pmatrix} 0 & 0 & -2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}}_c u^t = uCu^t = -1 \quad (14)$$

Schreibt man die algebraische Distanz um in

$$\epsilon_a(p_i, f(p, u)) = |f(p_i, u)| = |q_i u^t| \quad (15)$$

$$q_i = (x_i^2, x_i y_i, y_i^2, x_i, y_i, 1) \quad (16)$$

so erhält man das folgende Optimierungsproblem:

$$\begin{aligned} \min_u \sum_{i=1}^n [\epsilon_a(p_i, f(p, u))]^2 &= \min_u \sum_{i=1}^n [q_i u^t]^2 \\ &= \min_u \sum_{i=1}^n [(u q_i^t) * (q_i u^t)] \\ &= \min_u u \underbrace{\left[\sum_{i=1}^n q_i^t q_i \right]}_S u^t \\ &= \min_u u S u^t \end{aligned} \quad (17)$$

unter der Bedingung ((14)).

Die Minimierung wird durch das Lösen von

$$S u^t = \lambda C u^t \quad (18)$$

erreicht, was ein Generalisiertes Eigenvektorproblem darstellt. (18) enthält nur einen einzigen negativen Eigenwert, dessen Eigenvektor den Parametervektor u liefert, der eine Ellipse beschreibt.

Das gesamte Verfahren lässt sich in 6 Zeilen MATLAB Code ausdrücken, da sämtliche Rechenschritte als Funktionen im Programm enthalten sind:

```
1: D=[x.*x x.*y y.*y y ones(size(x))];
2: S = D' * D;
3: C(6,6) = 0; C(1,3)=-2; C(2,2)=1;C(3,1)=-2;
4: [gevec, geval] = eig(inv(S)*C);
5: [NegR,NegC] = find(geval<0 & ~isinf(geval));
6: a=gevec(:,NosC);
```

In den Zeilen 1 und 2 wird die Streumatrix S erzeugt, Zeile 3 erzeugt die Bedingung C . Die Eigenvektoren werden in Zeile 4 berechnet, aus denen dann in Zeile 5 der negative Eigenwert ausgelesen wird.

Da diese Vorgehensweise in bestimmten Situationen Fehler liefert, wurde sie von Halir und Flusser in [HF98] verbessert, um sie sowohl robuster, als auch schneller

zu machen.

Direktes spezifisches Ellipsenfitting hat den Vorteil einer schnellen Berechnung, die garantiert immer zu einer Ellipse führen wird, aber auch den Nachteil, nicht robust gegen Ausreißer zu sein. Dies ist an der Abb.26 dargestellt, die das Ergebnis von [HF98] an einer mit Paintshop erstellten Ellipse mit einer Störung zeigt. Durch die Verwendung des algebraischen, statt des euklidischen Abstandes tendiert das Verfahren zudem dazu, die Messpunkte im inneren der Ellipse zu bevorzugen und somit zu kleine Ellipsen zu generieren.

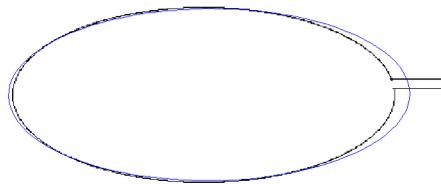


Abbildung 26: Bei Ausreißern liefern direkte Fittingmethoden Fehler

4 Implementierung

Im folgenden Kapitel wird die Implementierung zweier für die Aufgabe am geeignetsten erscheinenden Ellipsensuchverfahren aus Kapitel 3 beschrieben. Da diese Verfahren auf Kantenbildern basieren, wird hierzu im ersten Teil ein Verfahren vorgestellt, mit dem man aus den Eingangsbildern die Kantenbilder erzeugt. Im nächsten Schritt werden die Verfahren implementiert, um danach eine Anpassung auf den vorgestellten Anwendungsfall zu erhalten. Der letzte Schritt der Implementation ist die Anpassung auf die Radverfolgung.

4.1 Bildvorverarbeitung

Da die Ellipsensuche Kantenbilder benötigt, muss für die Suche nach Ellipsen eine Kantendetektion durchgeführt werden. Da jede dabei entstehende Kante eine Erhöhung der Rechenzeit bedeutet, entfernt eine Filterfunktion alle Kantenpixel aus dem Bild, die definitiv keine Ellipsenbestandteile sein können.

Die Implementierung Vorverarbeitungsschritte wird in den folgenden zwei Abschnitten beschrieben, unterteilt in die Kantenerzeugung und Filterung.

4.1.1 Kantenerzeugung aus einem Realbild

Der erste Schritt der Bildverarbeitung ist eine Rektifizierung des Eingangsbildes, um die durch die Kameralinse auftretende Verzerrung des Bildes zu korrigieren. Da diese durch das Bosch-Aufnahmesystem automatisch geschieht, wird hier nicht näher darauf eingegangen.

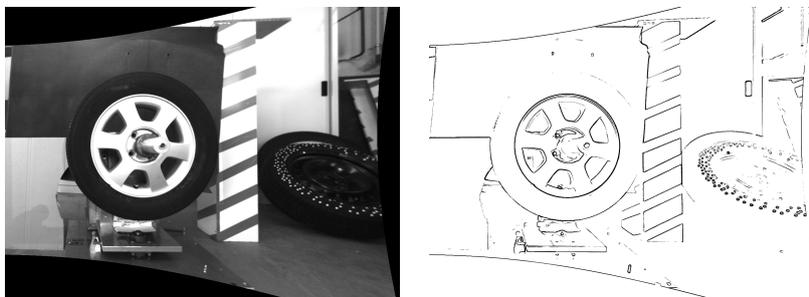


Abbildung 27: Ergebnis der Kantenerkennung
links: Startbild rechts: Nach Kantenerkennung

Der erste Schritt der Bildvorverarbeitung besteht in einer Aufbereitung des Kontrastes des im Graustufenformat vorliegenden Eingangsbildes um gut voneinander trennbare Kanten zu erhalten. Hierzu gibt es, wie in Kapitel 2.5.2 beschrieben, verschiedene Verfahren, aus denen die Contrast Limited Histogram Equalization ausgewählt wird, da diese unabhängig von der Lackfarbe des das Bild dominierenden Autos den Kontrast optimiert.

Da trotz CLAHE nach der Kontrastverstärkung ein Restrauschen vorhanden ist, wird das Bild mit einem Glättungsfilter weichgezeichnet. Aus den in Kapitel 2.5.3 vorgestellten Verfahren eignet sich der Medianfilter für diesen Anwendungsfall am

besten, da durch dessen Eigenschaften kleine Punkte im Bild verschwinden statt wie bei Faltungsoperatoren, wie z.B. dem Averagefilter, mit weichgezeichnet zu werden.

Die Kantenerkennung wird mit einem optimierten Sobeloperator nach Jaehne durchgeführt, da dieser einen geringeren Gradientenfehler als der Standardsobel aufweist, wie auch der Test in Kapitel 5.1.1 zeigt.

Um nach der Kantenerkennung nicht schwache Kanten, die meist nur erkanntes Bildrauschen und Störungen repräsentieren, in die Berechnungen einfließen zu lassen, werden mit einem Schwellwert alle zu schwachen Kanten aus dem Bild entfernt. Um manuelle Schwellwerte zu vermeiden, die schon dann nicht mehr gelten, wenn das nächste Bild eine andere Helligkeit hat, wird hier das Schwellwertverfahren nach Otsu verwendet.

Das Ergebnis der Kantenerzeugung ist in Abb.27 zu erkennen.

4.1.2 Filtern des Kantenbildes

Bei zusammenhängenden Figuren wie in Abb.28 wird zumindest bei den Fittingverfahren nur eine der beiden Ellipsen als solche erkannt. Daher müssen solche zusammenhängenden Objekte vor der Weiterverarbeitung in einzelne Teilobjekte segmentiert werden.

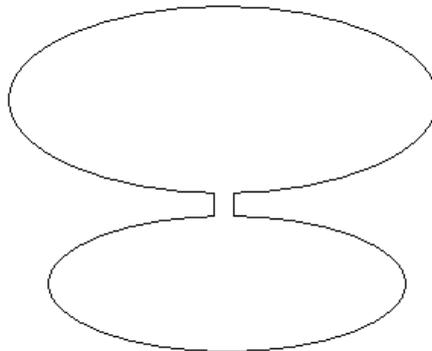


Abbildung 28: 2 Ellipsen in einer Kontur

Den ersten Schritt bildet hierfür die Auftrennung der Konturen an Verbindungsstellen in Grundobjekte wie Ellipsen, Bögen, oder Linien. Da solche Verbindungsstellen an den Konturen in den meisten Fällen durch Ecken gekennzeichnet sind, werden diese mit einem Eckendetektor gesucht und durch Herausnahme der Verbindungspixel voneinander getrennt. Um die Form der Grundelemente zu erhalten und diese nicht noch weiter zu zerteilen ist es wichtig, einen Eckendetektor zu nutzen, der eher zu wenig, als zu viele Ecken erkennt. Daher wurde der Erkennungsalgorithmus von He/Yung ([HY04]) gewählt, der im Gegensatz zu z.B. Harris diese Eigenschaft besitzt (siehe hierzu auch Abb.12 und Abb.13).

Die nun im Bild vorhandenen Grundelemente werden im nächsten Schritt auf die

Möglichkeit untersucht, aus ihnen Ellipsen bilden zu können und bei einem negativen Ergebnis als Störung aus dem Bild entfernt. Um hierbei das versehentliche Löschen eigentlicher Ellipsenteile zu verhindern dürfen nur diejenigen Regionen gelöscht werden, die definitiv keine Ellipsen bilden können, wobei selbst bei leichten Zweifeln an diesem Grundsatz eine Region bestehen bleibt.

Um als Ellipsenteil zu gelten, müssen die Objekte des Bildes zwei Grundbedingungen erfüllen:

1. Größe
Aus zu kleinen Elementen können keine korrekten Ellipsen berechnet werden (Siehe hierzu auch Kap.5.2.3). Daher müssen die Konturen im Bild eine Mindestgröße aufweisen.
2. Krümmung
Objekte unterhalb einer Mindestkrümmung werden als Linien betrachtet und deshalb aus dem Bild entfernt.

Die Aufteilung des Kantenbildes in einzelne zu berechnende Regionen wird mittels dem Operator *bwlabel* aus der Image Processing Toolbox durchgeführt, der ein „Union-Find“ ähnliches Labelverfahren nutzt. Die Größe eines Elementes wird eine einfache Abfrage der Pixelanzahl bestimmt:

```
for nrLabel = 1:labels
    region = find(labeled_img==nrLabel);
    if size(region) <= min_object_size;
        img(region)=0;
    end
end
```

Für die Krümmung einer Kontur werden die Gradienteninformationen der Kantensuche genutzt, da diese die Richtung einer Kante angeben. Die Differenz des kleinsten vom größten Gradientenwert gibt die Krümmung an:

```
kruemmung = max(max(phi(region))) - min(min(phi(region)));
```



Abbildung 29: Linie mit Kurvenende

Durch das alleinige Einbeziehen des niedrigsten und höchsten Peak ist dieses Verfahren zwar in der Ausführung sehr schnell, aber auch anfällig gegen Ausreißer im Histogramm. Zudem wird nicht beachtet, dass beim Übergang von 359 auf 0 Grad eine Krümmung von 360 Grad statt 1 Grad gemessen wird. Da diese nachteiligen Effekte allerdings nur verhindern können, dass ein falsches Objekt gelöscht wird und nicht dazu führen, dass ein richtiges Objekt entfernt wird, ist dieser Mangel tolerierbar. Sollte die Linie als Gerade mit Kurvenende vorliegen,

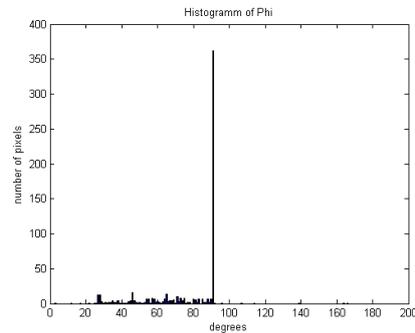


Abbildung 30: Histogramm der Linie mit Kurvenende

wie in Abb.29 zu sehen, wird sie durch diesen Berechnungsschritt nicht als Fehler erkannt, der Linienteil wird allerdings im nächsten Schritt entfernt.

Objekte, die durch die Eckenerkennung nicht vollständig in ihre Grundformen aufgeteilt werden konnten, wie das Beispiel in Abb.29, können durch die einfachen Regeln nicht erfasst werden. Daher muss bei ihnen die Krümmung der Objektbestandteile anders bewertet werden. Die bekannteste Methode, Objekte in Kurven und Geraden zu unterteilen, ist die Houghtransformation [Hou62] mit der Linien erkannt werden können. Diese Methode braucht allerdings einigen Rechenaufwand und findet nur gerade Linien. Daher wurde eine Methode entwickelt, die die aus der Kantenerkennung schon vorhandenen Gradienteninformationen ausnutzt.

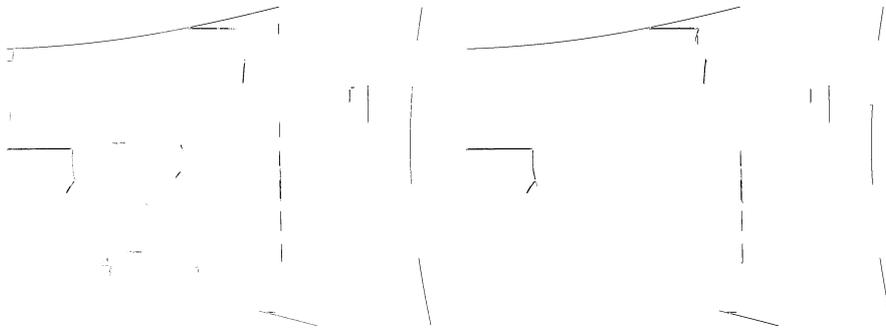


Abbildung 31: Die bei der Histogrammsuche als zu entfernend markierten Pixel (links) und die tatsächlich entfernten Pixel(rechts)

Die Ausreißer in einer Ellipsenform sind im Histogramm der Ellipse als starke Ausschläge zu erkennen. Je weiter diese Ausschläge vom Mittelwert aller Häufungen der Winkel entfernt liegen, desto eher sind sie einer nicht ellipsenartigen Form zuzuordnen und können gelöscht werden. Im Beispiel von Abb.30 ist dies an der Häufig des 90° Winkels zu erkennen, der die horizontale Linie darstellt. Löscht man hier alle Pixel, deren Gradientenwert 90° beträgt, so wird die Linie aus dem Bild entfernt.

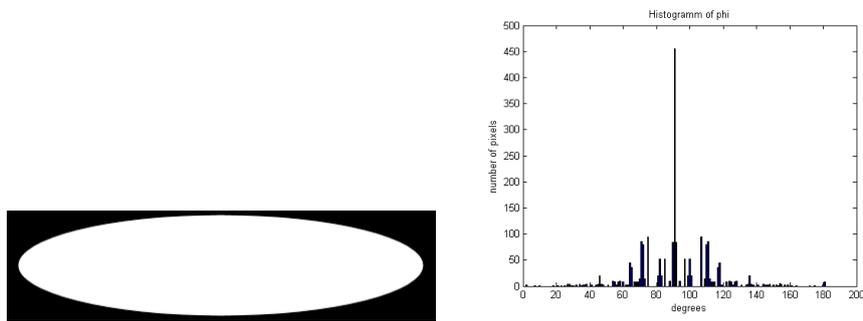


Abbildung 32: Gradientenhistogramm der links dargestellten schmalen Ellipse

Da stark gedehnte Ellipsen bei geringer Auflösung der Winkel einen ähnlich hohen Ausschlag wie eine Linie haben können, wie in Abb.32 gezeigt, muss ein vom Mittelwert abhängiger Schwellwert aufgabenabhängig gesetzt werden, der gleichzeitig das erlaubte Haupt-/Nebenachsenverhältnis der Ellipsen bestimmt. So kann sich z.B. ein Autorad nicht so weit drehen lassen, dass ein Verhältnis von 1:1.8 übertroffen wird². Im Beispiel von Abb.30 führt dieses Verfahren zur Löschung des Linienteils.

Da die Wahrscheinlichkeit hoch ist, dass man bei der Entfernung aller Pixel eines Objektes, die denselben Gradienten haben, Kantenpixel aus einer Ellipse löscht, werden die Pixel vor dem Löschen in ein temporäres Bild gespeichert (siehe Abb.31). Dieses Bild wird nochmals gelabelt und die einzelnen Label auf ihre Größe untersucht. Nur wenn sie eine Mindestgröße überschreiten werden sie entfernt.

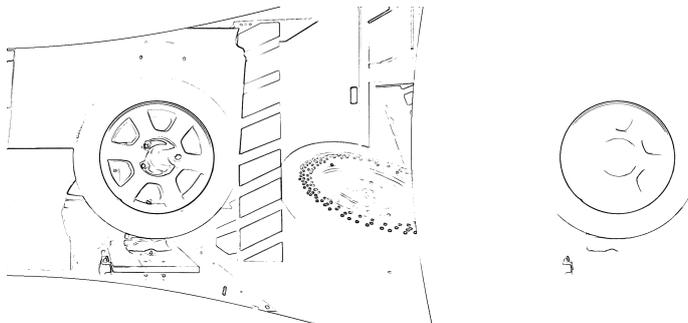


Abbildung 33: links: Nach der Kantenerkennung rechts: Nach dem Filtern

Um hierbei zu vermeiden, dass kleine Lücken in geraden Linien, z.B. durch eine fehlerhafte Tensorberechnung, die Linie unterbrechen und somit ein Löschen verhindern, wird eine Closingoperation durchgeführt. Dies ist eine einfache Möglichkeit Lücken zu schließen und verbessert die Erkennungsrate von fehlerhaften Linien merklich.

Nach dieser Operation werden die entfernten Ecken wieder in das Bild einge-

²Daten beruhen aus Auswertungen einer 7er BMW

setzt, um Objekte zusammenzufügen, die durch diese Operation getrennt wurden. Da im Anschluss wieder das Bild gelabelt und die Objekte nach Größe bewertet werden, führen neu auftauchende alleinstehende Ecken, die aus Objekten stammen die gelöscht wurden, nicht zu einer Verschlechterung des Bildergebnisses, sondern werden aus dem Bild entfernt.

Das Ergebnis der Objektsegmentierung ist in [Abb.33](#) zu sehen.

Zusammenfassung des Ablaufs

1. Kontrastverbesserung
2. Rauschentfernung mit Medianfilter
3. Kantenerzeugung mit optimiertem Sobelkantenfilter
4. Kontursegmentierung mit Eckenfilter
5. Filtern nach Größe und Krümmung
6. Filtern der restlichen Elemente durch Gradientenhistogrammfilter

4.2 Ellipsendetektion

4.2.1 Wahl der Verfahren

	Ausführungsgeschwindigkeit	Robustheit	Einfache Implementierung	Speicherbedarf	Genauigkeit	Gesamtpunkte
Gewichtung	1	2	1	1	4	
Hough						
Probabilistic Hough Transform	1	3	1	1		11
Randomized Hough Transform(RHT)	2	3	2	2		13
Restricted RHT (RRHT)	3	3	3	3		16
RRHT mit Konvexität	4	3	2	2		14
Fitting						
Ausgleichsrechnung	0	1	3	3		7
Bookstein	1	1	3	3		7
Taubin	1	1	3	3		7
Direktes Fitting	4	1	3	3		13
Direktes Fitting mit RANSAC	2	4	3	3		17

Tabelle 1: Einschätzung der einzelnen Verfahren aus der Literaturrecherche
 Je höher die Zahl, desto besser ist das jeweilige Verfahren in dieser Kategorie. Da die Genauigkeit erst experimentell evaluiert werden muss sind hier noch keine Werte eingetragen.

Aus den in Kapitel 3 vorgestellten Verfahren werden die beiden jeweils besten Vertreter der Houghtransformation und des Ellipsenfittings ausgewählt, um die verschiedenen Vorgehensweisen miteinander zu vergleichen. Das Hauptaugenmerk liegt hierbei auf der Genauigkeit und Robustheit der Algorithmen. Da die Houghtransformation sehr robust, aber nicht sehr genau ist, wird das hieraus gewählte Verfahren möglichst auf Genauigkeit optimiert, während das Verfahren aus den Fittingalgorithmen auf Robustheit optimiert wird.

Zur Auswahl des jeweils besten Verfahrens werden die einzelnen Verfahren in Tabelle 1 nach einzelnen Kriterien bewertet. Je höher der Wert eines Verfahrens bei einem dieser Kriterien ist, desto besser ist es in dieser Kategorie. Zusammen mit der Gewichtung der einzelnen Kriterien kann somit durch eine einfache Addition der unterschiedlichen Ergebnisse das beste Verfahren ermittelt werden. Einzig die Genauigkeit der Verfahren lässt sich nicht aus den Beschreibungen herauslesen, da in den unterschiedlichen Veröffentlichungen verschiedene nicht miteinander vergleichbare Testverfahren verwendet wurden. Die Genauigkeit muss somit in einem

Vergleich zwischen den beiden aus den anderen Kategorien ermittelten besten Verfahren verglichen werden.

Die beiden am besten bewerteten und somit zu implementierenden Verfahren sind die Restricted Randomized Hough Transformation (RRHT) und das Fittingverfahren mit RANSAC. Als Implementierungssprache für die Ellipsendetektoren dient MATLAB in der aktuellen Version 7.1. Im Gegensatz zu Programmiersprachen wie JAVA und C++ bietet MATLAB vielfältige Visualisierungs- und Debuggingmöglichkeiten. Dies und der Umstand, dass viele Standardverfahren der Bildverarbeitung schon implementiert sind, erleichtert die Erstellung eines neuen Algorithmus sehr. Ablaufpläne beider Algorithmen findet sich in Abb.38

4.2.2 Ellipsenerkennung mit Houghtransformation

Als Grundlage für die Ellipsenerkennung nach der Houghtransformation dient der Forschungsbericht über die Randomized Hough Transformation (RHT) mit zugehörigem Matlabcode von Samuel Inverso [Inv02], da dieser eine gute Struktur und Formelbeispiele für die weitere Implementation bietet. Der Nachteil dieser Implementation besteht darin, dass sie keine Winkelinformationen einbezieht und daher nur Ellipsen mit waagerechter Haupt-, oder Nebenachse berechnen kann. Da die Genauigkeitstests mit waagerechten Ellipsen durchgeführt werden können, reicht es allerdings für die Evaluation, welcher Algorithmus verwendet werden soll, erst einmal auf die Winkelinformationen zu verzichten. Aufgrund der Ergebnisse der Tests in Kapitel 5.1.2 und 5.2.1 wurde die Implementation der Winkelsuche nicht weitergeführt.

Die Erweiterung der RHT in eine Restricted RHT (RRHT) geschieht durch Labeling mittels des Operators *bwlabel* wie in Kapitel 4.1.

Die größte Schwierigkeit der Ellipsendetektion nach Hough ist die korrekte Bestimmung der Tangenten, die für die Mittelpunktbestimmung notwendig sind. Samuel Inverso schlägt hier vor, diese durch eine Ausgleichsgerade durch die umliegenden Punkte des gewünschten Punktes zu lösen. Der dazugehörige Code lautet:

```
nc = [ -10, 10, 10, -10 ];
nr = [-10 -10, 10, 10 ];

point = ind2sub(size(img), pointsIndex(1) );
[py, px] = find((roipoly(e,point(2)+nc, point(1)+nr)&e)>0);
[P,S] = polyfit(px, py,1);
```

Dabei bilden *nc* und *nr* ein 10*10 Pixel großes Polygon um den Punkt *point*. Innerhalb dieses Polygons werden alle Pixelwerte in die Matrix *[py, px]* gespeichert, die nicht der Hintergrundfarbe 0 entsprechen. Der Operator *polyfit* berechnet daraus die Ausgleichsgerade. *ind2sub* ist eine Hilfsfunktion, die aus den Matrixkoordinaten von Matlab kartesische Koordinaten erstellt und dazu die vorher berechnete Bildgröße, sowie eine Koordinate benötigt, die hier unter *pointsIndex* abgespeichert wurde.

Da diese Variante in manchen Fällen (siehe Ausreißer in Kapitel 5.1.1) eine falsche Gerade ausgegeben hat, wurde der Operator *Polyfit* gegen *fitline* aus der Toolbox von Peter Kovesi [Kov00] ausgetauscht. Da dieser Operator eine andere

Darstellung der ausgewählten Punkte benötigt, wird die Punktematrix transponiert übergeben:

```
TP = fitline([py, px]');
```

Ein Vergleich der beiden Tangentenvarianten ist in Abb.34 dargestellt, bei dem von Polyfit die rote Tangente falsch angelegt wurde, während sie von fitline besser, wenn auch nicht ganz korrekt, dargestellt wird.

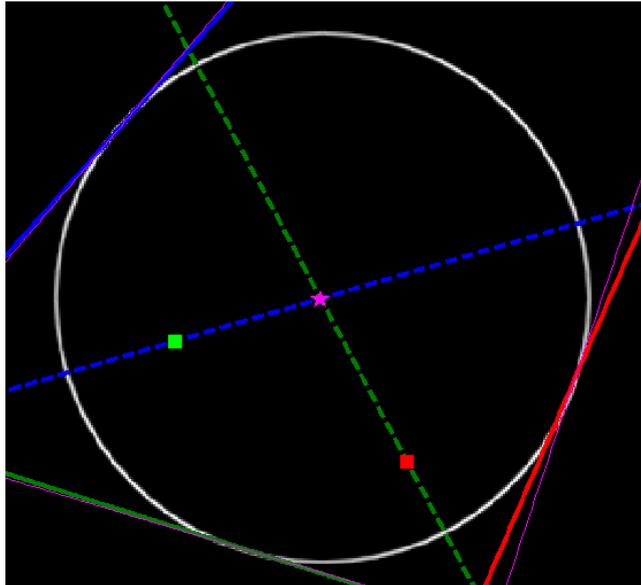


Abbildung 34: Tangenten mit polyfit (dick) und mit fitline(dünn)
Während polyfit die rote Tangente falsch anlegt zeigt fitline ein besseres Ergebnis

Beide verwendeten Verfahren, die Tangenten mit Ausgleichsgeraden zu bestimmen haben den Nachteil nicht für jede Ellipsengröße ein korrektes Ergebnis zu liefern, da sie einen festen Parameter benötigen, der die Polygongröße zur Auswahl der umliegenden Punkte festlegt. Daher wurde für einen Vergleich eine Variante implementiert, die aus der vor einer Ellipsendetektion bereits erfolgten Kantenerkennung die Gradienteninformationen für die zu untersuchenden Punkte ausliest.

```
pointphi = phi(pointsIndex(1))*180;
pointphi = pointphi+90;
TPM = -tand(pointphi);
TPB = I - TPM*J;
```

Da die Gradienteninformationen in der Implementation der Kantenerkennung auf einen Bereich von 0 bis 1 normiert wurden und 0 bis 180 Grad Drehung abdecken, müssen sie am Punkt *pointsindex* ausgelesen und als Gradzahl ausgegeben werden. Darauf werden 90 Grad addiert, da die Gradienteninformationen die Richtung der Kante angeben, die orthogonal zur Tangente steht. Der Tangens des so umgewandelten Gradienten liefert die Steigung, zu der dann der Ordinatenabschnitt ausgerechnet wird. Aufgrund der Implementierungsweise von

Matlab muss die Steigung negativ angegeben werden. Ein Vergleich der beiden Implementierungen der Tangentenbestimmung liegt in Kapitel 5.1.1 vor. Aufgrund dessen Ergebnisse wird in der weiteren Implementation die Tangentenbestimmung mittels Gradienteninformationen genutzt.

Der Schnittpunkt der Tangenten wird mittels der *Symbolic Math Toolbox* berechnet, die Gleichungssysteme mittels einfacher Befehle auflösen kann.

Zur Bestimmung der Haupt- und Nebenachse muss die Ellipse in den Ursprung transformiert und das Gleichungssystem 8 berechnet werden. Die Lösung des Gleichungssystems geschieht analog zur Bestimmung der Schnittpunkte der Linien mit der *Symbolic Math Toolbox*, mit der auch die Bedingung 9 überprüft wird. Besteht die Ellipsengleichung diesen Test, erfolgt eine Kontrolle, ob sie eine realistische Größe besitzt, indem die Größen der Achsen mit einem Schwellwert verglichen werden. Die endgültige Kontrolle, ob die Ellipse nicht nur rechnerisch, sondern auch im Bild existiert ist ein Einzeichnen einer Maske mit den Ellipsenparametern und eine Kontrolle, wie viele Punkte diese Maske abdeckt. Liegt dieser Wert unterhalb eines Schwellwertes, wird die Ellipse nicht gespeichert.

Die Parameter der gefundenen Ellipsen werden in einem Akkumulator gespeichert. Mittels Schwellwertparameter kann hierbei festgelegt werden, bei welcher Abweichung voneinander mehrere Parameter zu einem zusammengefasst werden, um zu verhindern, dass eine einzelne Ellipse durch viele Parameter abgebildet wird, die sich nur um Kommastellen voneinander unterscheiden.

4.2.3 Ellipsenerkennung mit Fitting

Das zweite Verfahren, ein Fittingverfahren mit RANSAC, setzt sich aus der Implementation des Ellipsendetektionsverfahren von Halir und Flusser aus [HF98] und einem RANSAC Algorithmus zusammen. Das Fittingverfahren ist schon in der Implementation von Halir und Flusser in MATLAB Code vorhanden und braucht daher nicht mehr geändert zu werden.

Da das Ellipsenfitting nicht robust gegenüber Ausreißern ist, der RANSAC Algorithmus dagegen allerdings nicht immer das optimale Ergebnis in Hinsicht auf die Genauigkeit liefert, wird der Algorithmus in 2 Schritten durchgeführt. Der erste Schritt dient dabei der Beseitigung der Messfehler aus der Menge der Punkte, der zweite Schritt generiert eine Ausgleichsellipse aus den verbliebenen Messpunkten.

Um im ersten Schritt die Ausreißer zu entfernen wird das Ellipsenfitting n -mal durchgeführt und das beste Ergebnis aus der Suche gewählt, wobei sich in Tests (siehe Kapitel 5.1) $n = \text{Pixelanzahl}/10$ als optimaler Parameter erwiesen hat. Durch die Übergabe nur der minimal nötigen Parameteranzahl von 5 Ellipsenpunkte an das Fitting ist die Wahrscheinlichkeit hoch, mindestens ein mal eine Ellipse aus dem Fit zu erhalten die nur Pixel aus der gesuchten Ellipse erhält.

Als Evaluierungsfunktion werden die von der jeweiligen Ellipse erfassten Punkte des Ausgangsbildes gezählt, wobei davon ausgegangen wird, dass die korrekte Ellipse die meisten Punkte eines Kantenbildes enthält. Dazu wird eine 1 Pixel breite Ellipse aus den Ergebnisparametern jedes RANSAC Durchlaufes gebildet und hieraus die Schnittmenge mit dem Originalbild gebildet. Die danach im Bild verbliebenen Pixel stellen das Qualitätsmaß dar. Damit eine fehlerhafte Ellipse die wesentlich größer als die gesuchte Ellipse ist nicht soviel Bildrauschen aufnimmt dass sie damit einen höheren Wert als die gesuchte Ellipse erlangt, werden alle Ellipsen mit der gleich

großen Anzahl von 200 Punkten im Bild eingezeichnet. Die höchstmögliche Punktzahl besteht somit in 200 im Bild verbleibenden Pixeln. Ein Beispiel für eine solche gebildete Ellipse bietet das mittlere Bild aus Abb.35

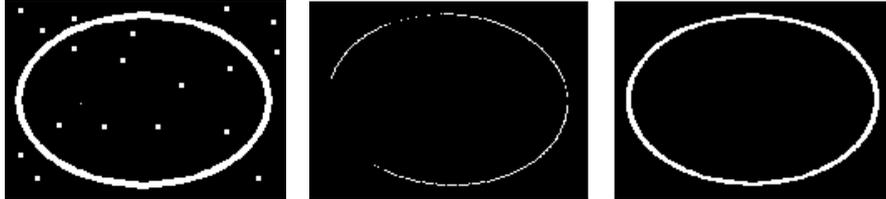


Abbildung 35: Von links nach rechts: Die Ausgangsellipse mit Störungen, eine Grafische Darstellung der Evaluationsfunktion und der Endsuchschlauch für die abschließende Ellipsendetektion

Im zweiten Schritt wird zuerst die beste Ellipse aus dem ersten Teil gesucht und aus dieser durch Dilatation ein 3 Pixel breiter Suchschlauch gebildet, der mit dem Originalbild verundet wird. Alle Punkte, die außerhalb dieses Suchschlauches liegen, werden als Messfehler aus dem Bild entfernt, während die innen liegenden Punkte komplett als Punktmenge an das direkte Fitting übergeben werden. Somit ist die Ellipsengleichung überbestimmt und es wird eine Ausgleichsellipse durch die Punkte gelegt, die die gesuchten Ellipsenparameter enthält. Anhand des Beispiel aus Abb.35, bei dem mit diesem Verfahren alle Störungen aus dem Bild genommen wurden, wird dies verdeutlicht. Aus dem rechten Bild wird hier die Ellipse gebildet.

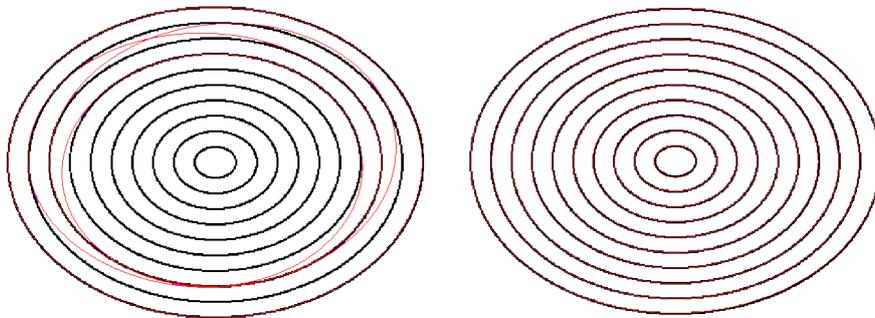


Abbildung 36: Suche auf nah beieinander liegenden Ellipsen. Links das Ergebnis mit der Suche über das ganze Bild und einer Anzahl von 10000 RANSAC Durchläufen, rechts das Ergebnis einer Suche über einzelne Region mittels Labelingverfahren bei 10 Labeln mit jeweils 100 RANSAC Durchläufen

Da das Fittingverfahren nur eine Ellipse auf einmal erkennen kann, eignet es sich in dieser Form noch nicht für die Suche mehrerer Ellipsen in einem einzelnen Bild. Für die dazu nötige Erweiterung gibt es zwei Möglichkeiten der Implementation. Bei der ersten Variante wird aus den Ellipsen des RANSAC Algorithmusses nicht nur die beste Ellipse, sondern n beste Ellipsen ausgewählt, aus deren Parametern abschließende Suchschläuche gebildet werden. Da es bei der Suche mit RANSAC

sehr häufig vorkommt, dass ein und dieselbe Ellipse an unterschiedlichen Stellen der Ellipse gefunden wird können die beste n Ellipsen alle aus der selben Ellipsengleichung stammen und müssen daher auf Gleichheit getestet werden. Da somit manuelle Schwellwerte gesetzt werden müssen, ab denen 2 Ellipsen nicht mehr als gleich gelten kann es hier bei einigen Einstellungen passieren, dass nahe beieinander liegende Ellipsen nicht mehr als getrennte Ellipsen gefunden werden können. Setzt man die den Schwellwert so, dass diese Ellipsen gefunden werden können, werden auch fehldetektierte Ellipsen, die eine hohe Punktzahl erhalten haben unter die besten Ellipsen gewählt. Ein weiterer Nachteil ist die nötige sehr hohe Anzahl von RANSAC Durchläufen, die für die Erkennung der Ellipsen nötig ist, da die Wahrscheinlichkeit gering ist, die 5 Parameter aus einer einzelnen Ellipse auszuwählen. Ein Beispiel hierfür bietet Abb.36, bei deren Ellipsen auch mit einer Anzahl von 10000 RANSAC Durchläufen nicht alle Ellipsen gefunden wurden. Der Vorteil dieses Verfahrens ist es, dass auch Ellipsen aus nicht zusammenhängenden Regionen gebildet werden können, wie es Abb.37 zeigt.

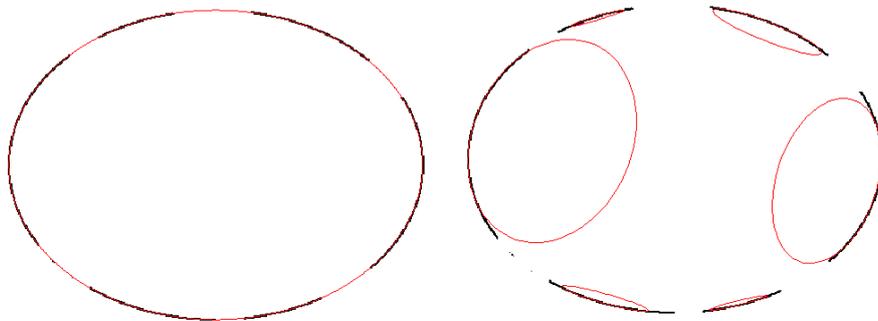


Abbildung 37: Suche auf einer unterteilten Ellipse. Links das Ergebnis des Suchverfahrens über das gesamte Bild, rechts das Ergebnis der Suche pro Region

Die zweite Variante teilt das Bild wie bei der RRHT in einzelne Regionen aus zusammenhängenden Pixeln auf und führt getrennt für jede Region ein Ellipsenfitting durch. Der Vorteil dieses Verfahrens ist die niedrige benötigte Anzahl von RANSAC Durchläufen, da die Wahrscheinlichkeit sehr hoch ist dass bei den 5 an das Fitting übergebenen Punkten alle Punkte auf derselben Ellipse liegen (siehe Abb.36). Der Nachteil dieser Variante besteht darin, dass keine Ellipsen gefunden werden können, die so in verschiedene Region geteilt sind, dass keine dieser Regionen eine Größe von über 60% der Gesamtellipse enthält. Siehe hierzu auch Abb.37 und den Test in Kapitel 5.2.3.

Implementiert wurden beide Verfahren. Das Verfahren zum Fitting pro Region ist das verwendete Standardverfahren zur Ellipsensuche. Die Suche über das komplette Bild wird in Kapitel 4.3.1 bei der Zusammensetzung von Objekten verwendet.

4.2.4 Vergleich der Verfahren

Beide implementierten Verfahren werden in Kapitel 5.1.2 miteinander verglichen, um zu entscheiden, welches von beiden für die weitere Implementation verwendet und auf den Anwendungsfall angepasst wird. Durch das Testergebnis ist die Entscheidung hierbei auf das Fittingverfahren gefallen, da dieses die wesentlich höhere

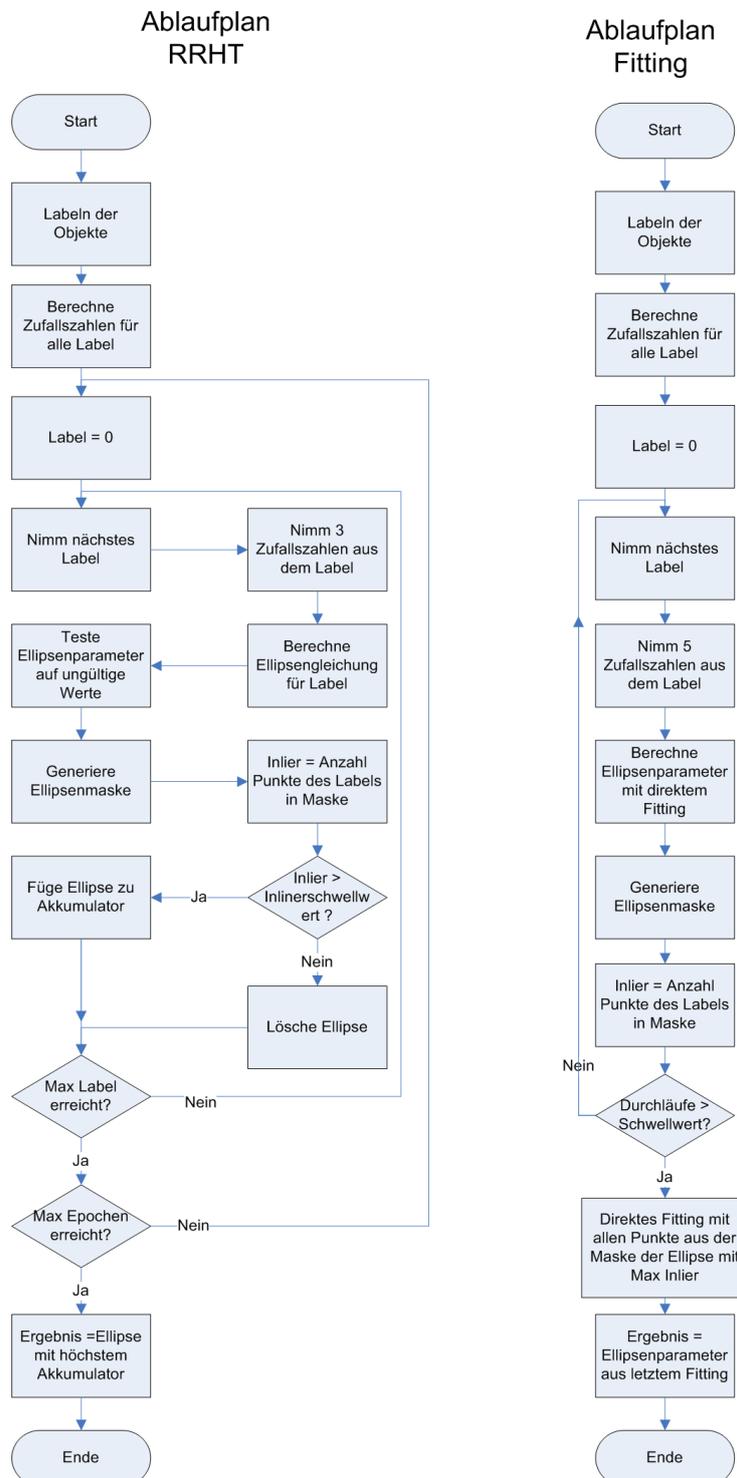


Abbildung 38: Ablaufpläne der Verfahren zur Ellipsendetektion

Genauigkeit aufweist.

Dieses Verfahren wurde daher in Kapitel 5.3 auf eine zufällige Auswahl von Radbildern angewandt, um die Besonderheiten echter Bilder im Vergleich zu den bisher verwendeten Testbildern zu evaluieren. Die dadurch gewonnenen Erkenntnisse werden dann in Kapitel 4.3 umgesetzt.

4.3 Erhöhung der Erkennungsleistung

Die Optimierung der Ellipsensuche auf den Anwendungsfall der Raderkennung erfordert eine Anpassung des Algorithmusses auf die Spezialfälle des Tests an Rädern in Kapitel 5.3. Dafür wird ein einfacher Algorithmus gesucht, der unterbrochene Konturen wieder zusammensetzen kann und ein weiterer Algorithmus, der aus zwei Stereobildern die zueinander passenden Ellipsen auswählt. Der durch die Optimierungen aus dem Anwendungsfall veränderte Gesamtalgorithmus wird in seiner Abschlussform im Abschnitt 4.3.4 beschrieben.

4.3.1 Zusammensetzen unterbrochener Linien

Unterbrochene Linien entstehen im Kantenbild wenn Teile einer schwachen Kante einen Grauwert unterhalb des Schwellwertes besitzen, der die Kanten vom Hintergrund trennt. Da dies in der Radsuche häufiger auftritt, besonders bei schwarzen Felgen auf schwarzem Grund, muss hier ein Algorithmus gefunden werden, der die gebrochenen Konturen wieder zusammensetzt. Da das Zusammensetzen von Konturen ein umfangreiches Themengebiet darstellt, dessen gesamte Thematik hier darzustellen zu aufwändig wäre, wird hier nur evaluiert, ob diese Aufgabe auch mit einem einfachen Verfahren gelöst werden kann.

Ein häufiges verwendetes Verfahren zum Zusammensetzen unterbrochener Konturen ist die auch in [JY04] genutzte Weiterführung der Kontur in Richtung des letzten Gradienten. Das Problem dieses Verfahrens ist es allerdings, dass sich die Fortsetzung der Kontur in dieser Richtung befinden muss, was schon bei einer Störung wie in Abb.39 nicht mehr gegeben ist. Auch wenn hier der Suchwinkel vergrößert wird, findet man möglicherweise nicht den weiteren Teil der Kontur. Zudem muss man bei dieser Verfahrensvariante alle Enden jeder Kontur auf mögliche Weiterführungen testen.

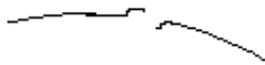


Abbildung 39: Beispiel einer Störung, die die Konturzusammensetzung erschwert

Eine andere Variante besteht darin, die schon vorhandenen Ellipsen des RANSAC für diese Suche und die Auswahl, welche Kontur mit welcher anderen Kontur getestet werden soll, einzusetzen. Bei den RANSAC Durchläufen einer Kontur ist die Wahrscheinlichkeit groß, mit einer der gezeichneten Ellipsen die Fortführung der Kontur zu treffen. Daher werden bei jedem RANSAC Durchlauf alle Konturen im

Bild, die von einer eingezeichneten Ellipse getroffen werden, darauf getestet, ob die aus beiden Konturen zusammen gewonnene Ellipse ein besseres Ergebnis als die Einzelellipse erzielt. Sollte dies eintreffen, wird beiden Konturen das kleinere ihrer Label zugeordnet, und die Labelsuche von diesem Punkt aus wieder fortgesetzt. Gehören alle Konturen des Bildes einer gemeinsamen Ellipse an, so werden somit alle zu einem Label verbunden. Der Ablaufplan des Algorithmusses ist in Abb.40 dargestellt.

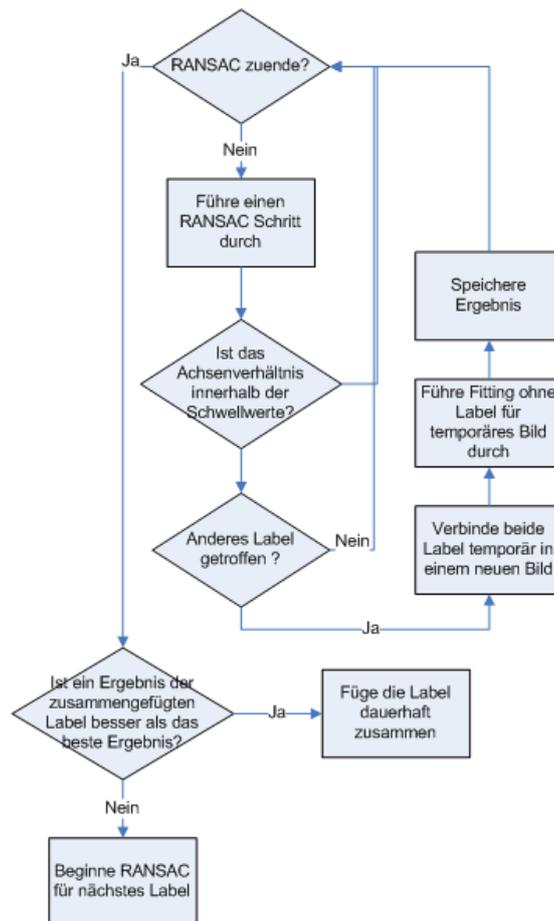


Abbildung 40: Ablaufplan der Objektverbindung

Eine Kontur die keine Ellipse darstellt, aber durch den Filter der Vorverarbeitung gelangt ist produziert viele Ellipsen die teilweise durch das ganze Bild gehen aber nur falsche Konturen treffen. Da durch diesen Effekt die Rechenzeit unnötig verlängert wird, ist vor der Überprüfung der zusammengeführten Label eine Abfrage eingebaut, ob das Achsenverhältnis innerhalb eines gewünschten Schwellwertes liegt. Kombiniert wird dies mit dem Vorwissen, dass die gesuchte Ellipse annähernd einen Kreis darstellt und somit das Verhältnis zwischen Haupt- und Nebenachse nicht zu stark von 1:1 abweichen darf.

Um überflüssige Doppelsuchen zu vermeiden, werden schon einmal getestete

Kombinationen nicht ein weiteres Mal überprüft, da es sonst vorkommen kann, dass innerhalb einer RANSAC Suche während jedes Durchganges die gleichen Kombinationen getestet werden. Diese Verhinderung gilt jedoch nicht für den Fall, dass eine Kombination noch einmal rückwärts gefunden werden kann, also z.B. die Label 1 und 3 erst erfolglos getestet werden, dann aber von 3 ausgehend noch einmal 1 berührt wird. Da es durch eine möglicherweise besser gelegene Position der zweiten Kontur vorkommen könnte, dass nur rückwärts die korrekten Konturen gefunden werden, wird die Doppelsuche in Kauf genommen.

Um eine Bevorzugung großer Ellipsen gegenüber kleineren Ellipsen zu vermeiden, wird durch die Ellipsengröße geteilt, um einen normierten Gesamtwert zu erhalten.

Der abschließende Vergleich des Ergebnisses der Ausgangskontur mit dem besten Ergebnis der kombinierten Konturen und der mögliche Zusammenschluss zu einem einheitlichen Label findet nach der abschließenden Bewertung der Kontur statt.

Eine weitere Möglichkeit der Zusammensetzung der Ellipsenteile ist das Fitting ohne vorheriges Labeln, wie es in Kapitel 4.2.3 beschrieben ist. Dies hat allerdings den schon beschriebenen Nachteil, dass die Suche in einem großen Bild sehr aufwändig sein kann und somit sehr viel Rechenzeit verbraucht. Für eine tolerable Rechenzeit sollte hier erst die Anzahl der Objekte im Bild stark verkleinert werden.

Die beschriebenen Verfahren werden in Kapitel 5.4.1 auf ihre Funktionsfähigkeit auf künstlich erzeugten Ellipsen und auf Radbildern getestet.

4.3.2 Binarisierung des Ausgangsbildes

Bei hellen Felgen ist die Entfernung der konzentrischen Kreise am Felgenrand einfacher als bei dunklen Felgen, da sich hier die Felge durch ihre helle Farbe erkennbar vom Hintergrund abhebt. So liegt es nahe, eine Schwellwertoperation durchzuführen, durch die das Bild in helle und dunkle Bereiche getrennt wird. Somit entsteht am Übergang zwischen Felge und Reifen eine einzelne deutliche Kante. Einen Vergleich dieses Verfahrens mit dem Originalalgorithmus zeigt Abb.41. Die Schwellwertoperation wird mit dem Otsuoperator durchgeführt, der in Kapitel 2.5.1 erklärt wird. Da nach der Binarisierung des Bildes bei der Kantenerkennung nur noch starke Kanten vorhanden sind, ist eine Schwellwertoperation des Kantenbildes dadurch nicht mehr nötig und wird aus dem Code entfernt. Ein Vergleich in Kapitel 5.5 wird zeigen, an welcher Stelle des Algorithmus die Binarisierung das beste Ergebnis erzeugt. Eine Möglichkeit ist es, die Binarisierung vor der Bildvorverarbeitung als gesonderten Schritt durchzuführen, eine zweite die, sie nach der Medianfilterung einzusetzen. Die Stelle nach der Medianoperation wurde gewählt, da hier die Kontrastoptimierung schon durchgeführt wurde und das dabei entstehende leichte Rauschen durch die Medianfilterung bereits geglättet ist.

4.3.3 Auswahl der gesuchten Ellipsen aus den Stereobildern

In den Grundlagen der Stereorekonstruktion in Kapitel 2.6.2 ist beschrieben, dass bei dem hier genutzten Stereonormalfall die Y-Koordinaten eines Elementes, das in beiden Bildern auftaucht, in beiden Bildern gleich ist. Daraus lässt sich ableiten,

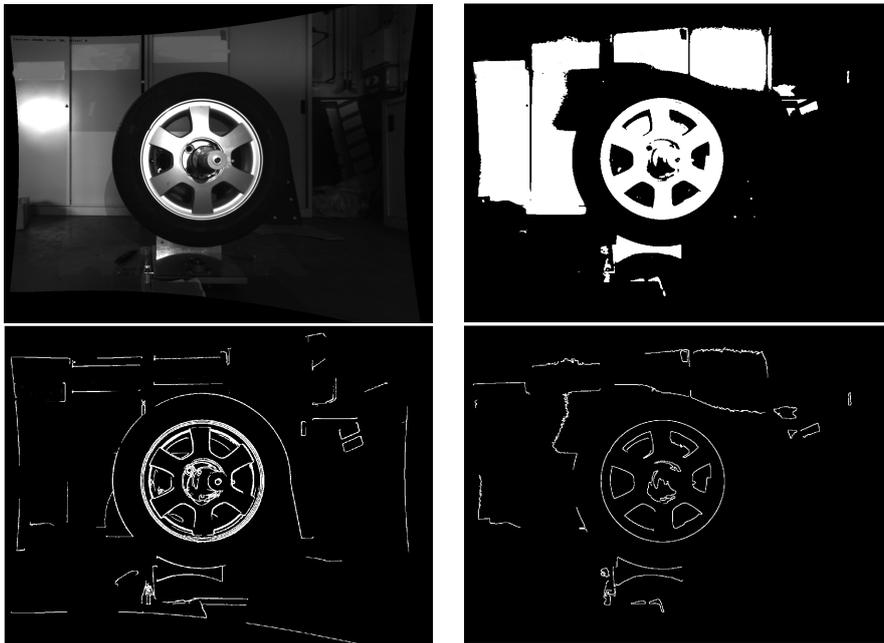


Abbildung 41: oben: Silberfelge vor und nach der Binarisierung
unten: Kantenerkennung mit Originalalgorithmus und mit Binarisierung

dass 2 Ellipsen aus den beiden Stereobildern, die beide im 3D Raum den gleichen Kreis darstellen, einen gemeinsamen Mittelpunkt auf der Y-Achse besitzen. Durch Vorwissen ist zudem bekannt, dass die gesuchten anfänglichen Startbilder eines am System vorbeifahrendes Rades nahezu kreisförmige Ellipsen aufweisen. Daher ist anzunehmen, dass auch die Haupt- und Nebenachse in beiden Bildern ähnlich sind. Für die Suche der zusammengehörigen Ellipsen werden die Parameter der in den Einzelbildern gefundenen Ellipsen daher auf diese Eigenschaften verglichen und Ellipsen ohne „Partner“ entfernt. Ein Ergebnis dieses Vergleichs findet sich in den Abbildungen 42 und 43.

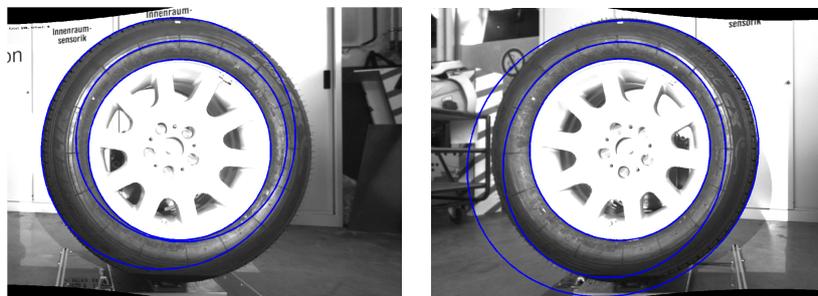


Abbildung 42: Ellipsen des linken und rechten Rades vor dem Vergleich

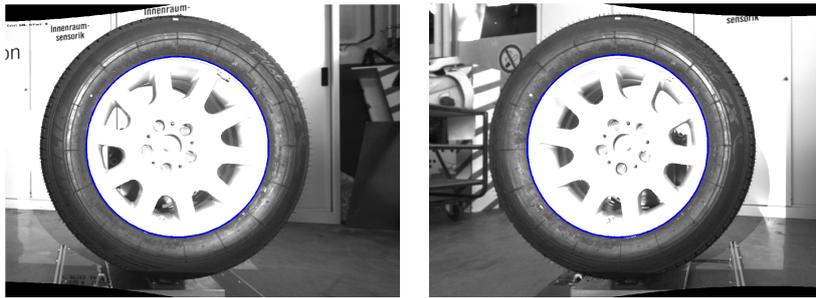


Abbildung 43: Ellipsen des linken und rechten Rades nach dem Vergleich

4.3.4 Zusammenfassung des Einzelraderkennungsalgorithmus

Die Endfassung des Einzelraderkennungsalgorithmus startet mit der Übergabe der Einzelbilder des Stereokamerasystems. Diese Bilder werden zuerst mit CLAHE kontrastoptimiert und danach vom Medianfilter weichgezeichnet. Auf diesen Bildern wird eine Binarisierung mit dem Otsuoperator durchgeführt, auf der danach die Kanten mittels eines optimierten Sobeloperators gesucht werden. Nach der nun erfolgenden Filterung der Kanten werden die Ellipsen des Bildes mittels des Fittingalgorithmus aus Kapitel 4.2.3 gesucht. Aus diesen wird das am besten zueinander passende Ellipsenpaar gewählt. Ein Ablaufplan des Algorithmus findet sich in Abb.44. Der Test dieses Verfahrens steht in Kapitel 5.5.

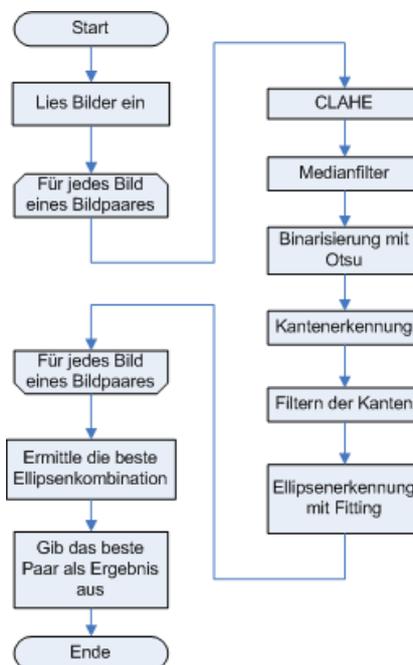


Abbildung 44: Die Endfassung des Algorithmus zur Einzelraderkennung nach der Optimierung auf den Anwendungsfall

4.4 Radverfolgung

Die Radverfolgung ist eine Einzelraderkennung, bei der Vorwissen über die vermutete nächste Ellipsenposition genutzt wird. Durch die Beschränkung der Ellipsensuche auf den Bildausschnitt der vermuteten Ellipse wird die Erkennungszeit deutlich gesenkt.

Das Schätzen der Position erfolgt durch die Beobachtung der Parameteränderungen der letzten n gefundenen Ellipsen. Der Mittelwert dieser Veränderungen stellt die geschätzte nächste Veränderung dar, die Standardabweichung der Veränderungen ist der zu erwartende Fehler dieser geschätzten Position. Liegt die Standardabweichung dabei über einem Schwellwert, wird sie als zu hoch angesehen und damit die Ellipse als Messfehler. Sollte die Standardabweichung auch dann noch zu hoch sein, wenn die als Messfehler deklarierte Ellipse nicht berücksichtigt wird, dann wird die nächste Ellipse wieder aus dem gesamten Bild gesucht.

Zur Berücksichtigung des zu erwartenden Fehlers der geschätzten Position wird der Bildausschnitt nicht auf die Maße der erwarteten Ellipsengröße beschränkt, sondern um die Größe des akkumulierten Fehlers aller Parameter in allen Richtungen erweitert. Der Ablaufplan des Algorithmusses findet sich in Abb.45

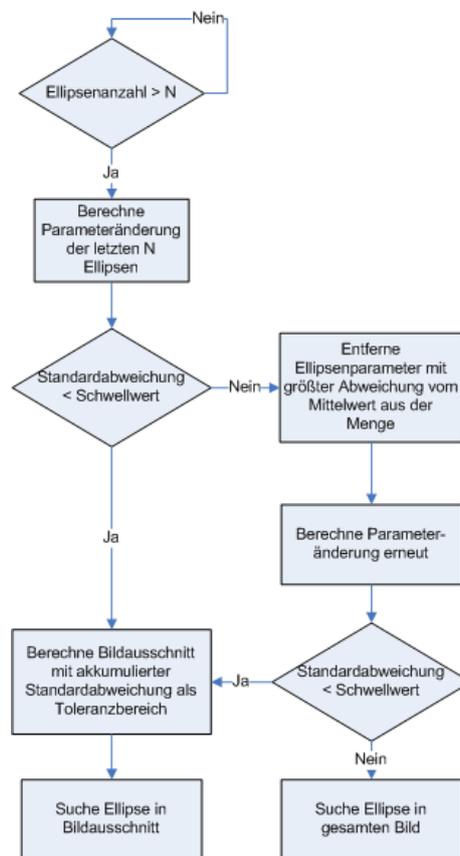


Abbildung 45: Ablaufplan der Radverfolgung

4.5 Berechnung der Tiefeninformationen

Die Berechnung der Tiefeninformationen der einzelnen Punkte des 3D Kreises kann durch eine Stereorekonstruktion nach dem Schema in Kapitel 2.6.2 erfolgen. Die Korrespondenzsuche, also die Suche nach der gemeinsamen Abbildung eines 3D Punktes in den 2D Bildern, stellt hier kein Problem dar, da die Koordinaten der Ellipsenpunkte bekannt sind.

Die aktuelle Forschung in der Stereorekonstruktion besteht nicht aus der Berechnung der Tiefeninformation, sondern bei der Lösung des Korrespondenzproblems. Da dieses, wie oben erwähnt, hier nicht auftritt, da die zueinander gehörenden Punkten auf derselben Y-Koordinate liegen und somit durch die Ellipsengleichung (z.B. (10)) berechnet werden können, wird hier nur die Möglichkeit einer 3D-Rekonstruktion vorgestellt und nicht näher darauf eingegangen.

Mit der Formel $Z = (B * C) / (x1 - x2)$; kann nun die Tiefe des Punktes bestimmt werden, wobei B der Basisabstand der Kameras ist, C die Kamerakonstante, x1 die X-Koordinate des Punktes im linken Kamerabild und x2 die X-Koordinate des Punktes im rechten Kamerabild. Die X-Koordinate des 3D Punktes erhält man durch $X = Z / C * x1$ und die Y-Koordinate durch $Y = Z / C * y$

5 Experimente

5.1 Evaluation verschiedener Implementierungsdetails

Während der Implementationsphase mussten einzelne Tests zur Wirkung verschiedener Parametereinstellung und zur Qualität der Algorithmen durchgeführt werden, die der Übersicht halber in diesem Kapitel aufgelistet werden.

Als Testellipse wird eine mit Paintshop Pro 7 erstellte Ellipse mit den Parametern $Mittelpunkt = (250/250)$, $Hauptachse = 200$, $Nebenachse = 150$, $\phi = 0$ genutzt. Da die Parameter unterschiedliche Maße (Pixel und Grad) nutzen, die nur sehr schlecht miteinander vergleichbar sind, werden sie nicht direkt mit den errechneten Parametern verglichen. Stattdessen wird zur Bewertung die Referenzellipse auf eine Linienstärke von 3 Pixeln erweitert und die Anzahl der Punkte der ermittelten Ellipsen summiert, die auf dieser Linie liegen. Als genügend genau erkannt gilt eine Ellipse, bei der 90% der Punkte von der Referenzellipse bedeckt werden. Dieses Verfahren ermöglicht den direkten Vergleich der ermittelten Ellipsen und gibt auch bei nicht genau erkannten Ellipsen an, wie viel Prozent der Ellipse korrekt sind.

Da beide implementierten Ellipsenverfahren mit Zufallswerten arbeiten, wird für ein statistisch aussagekräftiges Ergebnis das Erkennungsverfahren bei jedem Test 50x wiederholt. Das Ergebnis ist eine Auflistung, wie viel Prozent der 50 Testdurchläufe die Kriterien einer gültigen Ellipsenerkennung erfüllt haben. Die Tests sind, soweit das Ergebnis nicht nur aus einer Tabelle besteht, in der Reihenfolge Beschreibung, Ergebnis, Deutung aufgebaut.

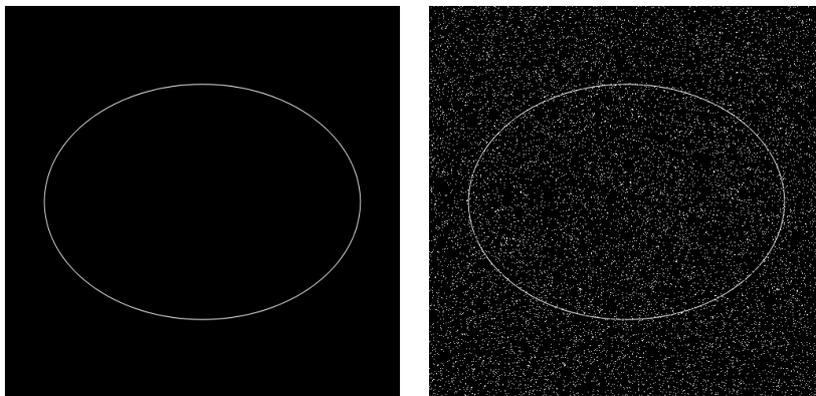


Abbildung 46: links: Referenzellipse rechts: Mit 10% Rauschen

5.1.1 Tangentenbestimmung bei der Houghtransformation

Beschreibung

Zur Bestimmung des Winkelfehler der beiden in Kapitel 4.2.2 vorgestellten Verfahren zur Tangentenbestimmung eines Ellipsenpunktes werden mittels eines vorher definierten Kreises die Lote zu den Tangenten am Tangentenpunkt berechnet. Bei einer korrekten Bestimmung der Tangenten müssen diese Geraden exakt durch den Mittelpunkt des Kreises verlaufen. Der Abstand der Geraden vom Mittelpunkt stellt

somit das Fehlermaß dar.

Ergebnis

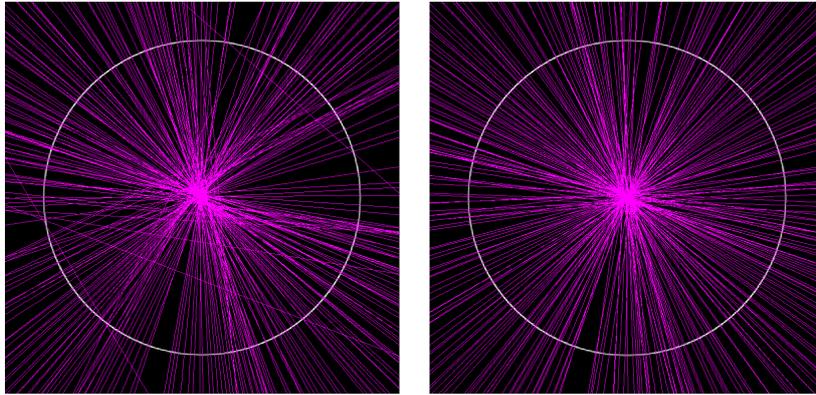


Abbildung 47: Winkelfehler der Verfahren zur Tangentenbestimmung
links: Fitting der umliegenden Punkte mit fitline
rechts: Gradientennutzung

Zur Veranschaulichung des Ergebnisses wurde jede 20. Gerade des jeweiligen Verfahrens in Abb.47 eingezeichnet. So sind auf dem Ergebnisbild der Fehlerbestimmung beim Fittingverfahren deutlich Ausreißer zu erkennen.

Da der Kreisradius bekannt ist, lässt sich hieraus leicht der Winkelfehler in Grad $\arctan(\text{Abweichung}/\text{Radius})$ ablesen. Bei der Tangentenbestimmung mit Fitting beträgt er 5.42° , bei der Bestimmung mit Gradienteninformation 1.43° . Bei der Bestimmung der Gradienteninformationen wurde ein optimierter Sobel nach Jähne ([Jäh02] Kapitel 12.5.5) verwendet. Mit einem Standardsobel beträgt der Winkelfehler 2° . Ein zweiter durchgeführter Test mit einer Verdünnung der Kreislinie durch eine Non-Maximum-Suppression zur Untersuchung der Auswirkung der Kantendicke auf die Tangentenbestimmung ergibt einen Winkelfehler für die Fittingvariante von 1.78° und für die Gradientennutzung von 1.49° (optimierter Sobel) bzw. 2.1° (Sobel).

Deutung

Der hohe Winkelfehler der Bestimmung mit Tangenten im ersten Testbild ist aus den Ausreißern erklärbar, die ein Zeichen dafür sind, dass dieses Verfahren nicht immer einsetzbar ist. So können manche Punkte im Kantenbild so ungünstig von anderen Punkten umgeben sein, dass sich durch Fitting nur sehr schlecht die korrekte Tangente bestimmen lässt. Das Testergebnis nach einer Non-Maximum-Suppression hat hierzu die Bestätigung geliefert, da die Ausreißer nicht mehr auftraten. Allerdings zeigt diese Anfälligkeit für Ausreißer, dass das Verfahren so nicht einsetzbar ist.

Die Kantenbestimmung durch Gradientennutzung ist wesentlich weniger anfällig für Ausreißer und zeigt ein relativ gutes Ergebnis für den Winkelfehler. Auffällig ist hierbei allerdings, dass der Winkelfehler bei weitem nicht so niedrig ist wie von Jähne in [Jäh02] Kapitel 12.5.5 prognostiziert. Durch das gute und robuste Bestimmen der Tangente wird das Verfahren mit Gradientennutzung für die

weiteren Tests genutzt werden.

5.1.2 Evaluation der Erkennungsgenauigkeit

Beschreibung

Zur Bestimmung der erreichbaren durchschnittlichen Genauigkeit der Houghtransformation und des Fittings werden die Algorithmen auf die Testellipse angewandt.

Ergebnis

Die beispielhaften Ergebnisse in Abb.48 zeigen die Erkennungsqualität der

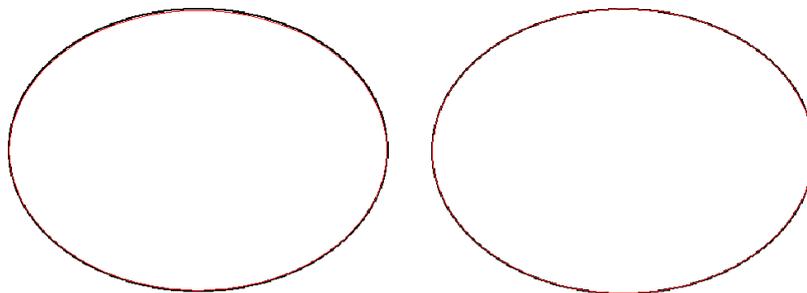


Abbildung 48: Vergleich der Ellipsenverfahren

links: Ergebnis der Houghtransformation bei 65% Genauigkeit

rechts: Ergebnis des Fittingverfahrens bei 100% Genauigkeit

Verfahren. Die Restricted Randomized Hough Transformation erreicht bei 50 Testdurchläufen eine durchschnittliche Genauigkeit von 62% und bei nur 2% der Durchläufe deckt sie über 90% der gesuchten Ellipse ab. Das Fittingverfahren wiederum erreicht eine durchschnittliche Genauigkeit von 100%.

Deutung

Das sehr schlechte Ergebnis lässt sich anhand der Abhängigkeit der Tangentenbestimmung von einer korrekten Winkelerkennung erklären, die im besten Falle wie oben gezeigt noch immer einen Fehler von 1.5° pro Tangente enthält. Da für die Mittelpunkterkennung drei Tangenten benötigt werden, akkumuliert sich dieser Fehler. Die Houghtransformation ist somit zwar für eine schnelle und robuste Ellipsenerkennung brauchbar, entspricht aber nicht den geforderten Genauigkeitsanforderungen. Das Fittingverfahren zeigt ein sehr genaues Ergebnis bei schneller Ausführungszeit.

5.2 Evaluation der Verfahrensqualität

Nachdem die Houghtransformation als mögliches Verfahren durch die mangelnde Genauigkeit ausgeschieden ist, wird das Fittingverfahren auf die optimalen Parametereinstellungen getestet. Dies umfasst eine Untersuchung, ab welcher Anzahl von RANSAC Durchläufen mit einem im Sinne der genauen Ellipsensuche optimalen Ergebnis zu rechnen ist, sowie die Ermittlung der nötigen Mindestgröße einer Ellipse zu ihrer pixelgenauen Bestimmung.

5.2.1 Robustheit bei Rauschen

Beschreibung

Dieser Test dient der Ermittlung, wie viele RANSAC Durchläufe bei welcher Anzahl an Fehlerpixeln benötigt werden, um die nahezu 100%ige Wahrscheinlichkeit einer korrekten Ellipsenerkennung zu erreichen. Für die Bestimmung der Testbedingungen dient die Bestimmung der Durchläufe mit der Wahrscheinlichkeitsrechnung.

Für einen einzelnen RANSAC Durchlauf werden 5 Parameter benötigt. Die Wahrscheinlichkeit P , mit einem Pixel p einen Punkt der Ellipse zu treffen und nicht einen Punkt aus den Messfehlern beträgt $P_{\text{einzelnerPixel}} = \frac{\text{AnzahlEllipsenpixel}}{\text{AnzahlGesamtpixel}}$. Für die 5 Parameter beträgt sie $P = \left(\frac{\text{AnzahlEllipsenpixel}}{\text{AnzahlGesamtpixel}}\right)^5$. Die für eine 99%ige Erkennung nötige Anzahl von Durchläufen D berechnet sich somit aus der Formel:

$$D = \ln(0.01) / \ln(1 - P) \quad (19)$$

Für ein Verhältnis von gültigen Messwerten zur Gesamtzahl von Messwerten von $1 : i$ ergibt diese Formel die Grafik aus Abb.49.

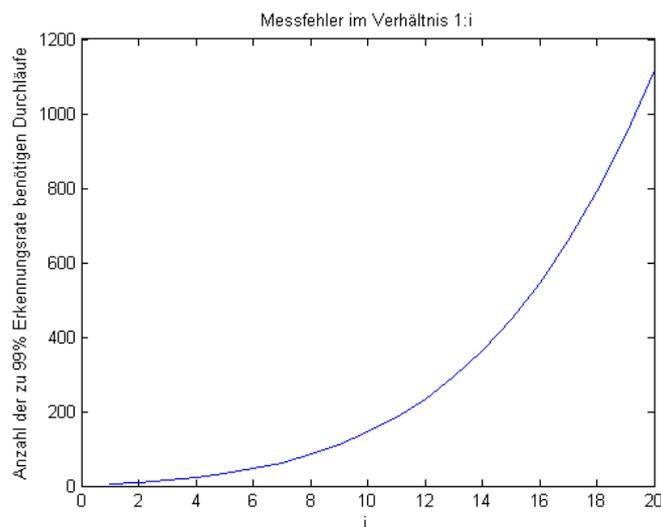


Abbildung 49: Verhältnis der benötigten Anzahl der Durchläufe zur 99%igen Erkennung zu der Anzahl der Messfehler

Zur Verifizierung dieses Ergebnisses durch einen Test des Algorithmusses wird die oben erwähnte Ellipse verwendet. Die Fehlerpixel werden dabei mit dem MATLAB Operator „imnoise“ und der Einstellung „Salt & Pepper“ generiert. Dieser erzeugt zufällig verteilte schwarze und weiße Pixel auf dem Testbild in einstellbarer Anzahl (siehe Abb.46). Als zu testender Bereich wird eine Anzahl von 0-200 Durchläufen gewählt, da die Laufzeit des Algorithmusses bis zu dieser Anzahl noch vertretbar ist. Das Rauschen des Bildes wird schrittweise um 0.5% Schritte von 0 und 10% erhöht, wobei die Prozentangabe die Anzahl der Bildpixel angibt, die mit schwarzen Pixeln bedeckt werden. Die Anzahl der RANSAC Durchläufe wird in 25er Schritten erhöht. So erhält man als Ergebnis eine zweidimensionale

Tabelle, die die Erkennungsgüte für jedes der gewählten RANSAC-Durchläufe / Noise Paare angibt. Die Ellipse enthält 2394 Pixel, ein Prozent Rauschen entspricht ungefähr 600 Messfehlerpixeln, also ca. 25% der Ellipsegröße.

Ergebnis

runs	noise						
	0%	0.5%	1%	1.5%	2%	2.5%	3%
25	100	96	78	46	24	18	8
50	100	100	92	72	40	38	14
75	100	100	98	80	60	44	28
100	100	100	100	88	78	50	32
125	100	100	100	94	72	62	42
150	100	100	100	96	86	70	46
175	100	100	100	96	80	72	40
200	100	100	100	96	94	78	52

Tabelle 2: Durchläufe mit über 90% Referenzübereinstimmung

runs	noise						
	0%	0.5%	1%	1.5%	2%	2.5%	3%
25	100%	98%	89%	65%	52%	44%	31%
50	100%	100%	96%	84%	64%	67%	46%
75	100%	100%	99%	92%	78%	70%	56%
100	100%	100%	100%	95%	89%	71%	64%
125	100%	100%	100%	97%	87%	80%	69%
150	100%	100%	100%	97%	92%	87%	73%
175	100%	100%	100%	98%	92%	89%	71%
200	100%	100%	100%	99%	97%	90%	77%

Tabelle 3: Durchschnittliche Übereinstimmung mit Referenzellipse

Die Testergebnisse sind in Tabelle 2 aufgeführt. Angegeben sind die Prozentzahlen der Durchläufe, die bei einer Maskenbreite von 3 Pixeln mindestens 90% der Pixel der Referenzellipse einschließen.

Deutung

Das Testergebnis bestätigt die in der Theorie gezeigte logarithmische Erkennungsleistung. Das Testergebnis ist bis zu einem Rauschanteil von 1% sehr gut und bei 1.5% Rauschen noch zufriedenstellend. Insbesondere kann durch dieses Ergebnis gezeigt werden, dass selbst bei einem Verhältnis von Ellipsenpunkten zu Messfehlern von 1.3:1 die Erkennung noch zuverlässig ist. Von diesem Testergebnis ausgehend wird eine Anzahl von $\text{Pixelanzahl}/10$ RANSAC Durchläufe für die sichere Erkennung einer Ellipse empfohlen. Von einem festen Wert wird abgeraten, da die Größe einer zu erkennenden Ellipse schwanken kann und daher bei einer festen Anzahl von Durchläufen die Anzahl bei einer zu großen Ellipse zu klein sein kann.

runs	Größe									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
25	100	100	99	95	89	84	73	49	35	45
50	100	100	99	99	96	96	92	63	59	44
75	100	100	100	99	99	96	95	80	61	51
100	100	100	100	100	100	97	97	79	68	59
125	100	100	100	100	100	100	96	92	75	60
150	100	100	100	100	100	100	94	86	65	66
175	100	100	100	100	100	100	99	90	74	68
200	100	100	100	100	100	100	100	86	72	64

Tabelle 4: Robustheit gegenüber einer zusammenhängenden Störung in Prozent

5.2.2 Robustheit bei zusammenhängender Störung

Beschreibung

Da der Fittingalgorithmus als Evaluationsfunktion des RANSAC die Anzahl der Pixel zählt, die von einer Ellipse erfasst werden, präferiert dieses Verfahren zusammenhängende Pixel in Ellipsenform. Es lässt sich somit von einer Störung immer stärker beeinflussen je ellipsenförmiger diese ist, wobei bei 2 zusammenhängende Ellipsen jeweils die größere berücksichtigt wird. Um diese Theorie in der Praxis zu evaluieren wird neben die Testellipse eine Ellipse gleicher Form, aber unterschiedlicher Größe projiziert, wobei diese Größe in Prozent der Originalellipse gemessen wird. Je mehr sie sich dabei der Testellipse annähert, desto stärker wird vorraussichtlich der Störfaktor werden, bis bei Größengleichheit der Ellipsen die Verteilung der gefundenen Ellipse bei ca. 50% auf beide Ellipsen liegen sollte.

Deutung

Das Ergebnis aus Tabelle 4 bestätigt die Vermutung, dass bei der für die Erkennung sehr ungünstigen Störung die Erkennungsrate des Algorithmus gegen 50% strebt, je mehr die Störung die Größe der eigentlichen Ellipse erreicht.

5.2.3 Bestimmung des nötigen Ellipsenanteils zur korrekten Erkennung

Beschreibung

Um den nötigen Anteil einer Ellipse zu ermitteln, der mindestens vorhanden sein muss, um eine Ellipse korrekt zu berechnen, wird die Ellipsenerkennung auf Teilstücke von 10-70% angewandt, die in 10% Schritten vergrößert werden.

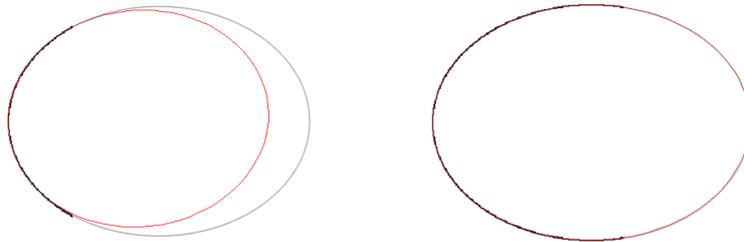


Abbildung 50: links: 20% der Ellipse , rechts: 60% der Ellipse
Der eigentliche Ellipsenverlauf ist in Grau, die erkannte Form in rot dargestellt

Ergebnis

Ab einer Teilstückgröße von 50% der Ursprungsellipse funktioniert die Ellipsendetektion mit mindesten 75 RANSAC Durchläufen sehr zuverlässig. Wie Abb.50 zeigt, werden zwar auch schon mit einer niedrigen Prozentzahl korrekte Ellipsen erkannt, allerdings verfügen diese nicht über die gewünschte Genauigkeit.

runs	Teilstückgröße						
	10%	20%	30%	40%	50%	60%	70%
25	0	0	4	56	86	88	98
50	0	0	8	52	92	94	100
75	0	0	14	66	98	100	100
100	0	0	4	62	98	98	98
125	0	0	2	60	92	94	100
150	0	0	2	62	98	96	98
175	0	0	0	66	98	100	98
200	0	0	2	82	98	100	100

Tabelle 5: Durchschnittliche Übereinstimmung mit Referenzellipse

Deutung

Die nötige Ellipsengröße liegt an der Diskrepanz zwischen subpixelgenauer Ellipsenparameter und nur pixelgenauer Teilellipsen. So gibt es bis zu einer Ellipsengröße von ca. 50-60% viele Ellipsen, die zwar die Pixel der Teilellipse abdecken, aber nicht die komplette Ellipse nachbilden.

5.2.4 Fazit der Evaluation an künstlichen Ellipsen

Es hat sich gezeigt, dass das Fittingverfahren zur Erkennung einer Ellipse sehr robust ist und ab einem Anteil von 50% der Ellipse ein sehr gutes Ergebnis zeigt. Ein Bildrauschen wie es z.B. von einer CCD Kamera bei einer starken Erhöhung der inneren Verstärkung erscheint (siehe Kapitel 2.6.1) wird bis zu einem Rauschanteil von 1-1.5% kompensiert. Objekte, deren Kante mit der Kante der gesuchten Ellipse verbunden ist, können im schlechtesten Fall einer Formübereinstimmung mit der gesuchten Ellipse bis zu 60% der Ellipsengröße besitzen, ohne sich störend auf die Suche auszuwirken.

Im Gegensatz dazu ist die Erkennung mittels Houghtransformation an der Genauigkeitsanforderung gescheitert. Sie ist zwar laut Literatur ein robustes Verfahren, da allerdings die Tangenten, die zur Bestimmung des Mittelpunktes nötig sind, nicht genau bestimmt werden können ist das Verfahren somit vorerst nicht für eine Anwendung mit sehr hoher Genauigkeitsanforderung geeignet.

Die Experimente an künstlichen Ellipsen haben somit gezeigt, dass es zumindest in der Theorie ein Verfahren gibt, das robust und genau genug für die gestellte Anforderung ist. Weitere Tests mit realen Bildaufnahmen müssen nun zeigen, ob diese Anforderungen auch in der Praxis erreicht werden können.

5.3 Evaluation der Besonderheiten von Radbildern

Beschreibung

Für die Erkennung von Autorädern muss herausgefunden werden, welche Arten von Felgen mit welcher Beleuchtung der Algorithmus definitiv nicht erkennen kann und bei welchen Arten es noch Probleme bei der Suche gibt. Zu diesem Zweck wird hier eine Versuchsreihe durchgeführt, in der der Algorithmus auf 770 zufällige Testbilder angewandt wird. Die Bilder wurden alle mit dem Testsystem aus Kapitel 2.6 erstellt. Das Testergebnis ist in Tabelle 6 aufgelistet.

Bildordner	Anzahl Bilder	erkannt	Verhältnis
dunkle Felge entfernt	154	2	4,38%
" (hell beleuchtet)	140	38	27,14%
dunkle Felge nah	58	0	0%
Strukturlicht	108	0	0%
helle Felge	304	263	86,51%
Fahrzeug mit heller Felge	6	4	66,67%
gesamt	770	308	40%

Tabelle 6: Testergebnisse der Ellipsenerkennung ohne Optimierungen auf den Anwendungsfall

Ziel der Erkennung ist es, den Übergang der Felge auf den Reifen als Ellipse darzustellen. Als Qualitätsmaß dient in diesem Testfall nur der Augenschein, da die Störungen detektiert werden sollen, durch die das Ellipsenfitting nicht in der Lage war, eine korrekte Ellipse zu erkennen. Da die Genauigkeit das wichtigste Maß dieser Arbeit darstellt, bedeutet Augenschein in diesem Falle, dass sich zwischen der angezeigten Ellipse und dem Rad, bzw. der Felge keine erkennbare Lücke zu sehen ist.

Das beste Ergebnis im Test erhielt eine helle Felge auf einem dunklen Grund wie



Abbildung 51: Testergebnis einer hellen Felge auf dunklem Reifen. Im linken Bild wurde der Übergang exakt erkannt, im rechten Bild ist ein leichter Fehler erkennbar

in Abb.51 zu erkennen, bei der sich die Kanten sehr gut abzeichnen. Von den 304 Testbildern auf diesem Reifen wurden 263 mit der vorher festgelegten Genauigkeit gefunden. Bei den restlichen 8 Bildern stimmte die Genauigkeit nicht, da an einzelnen Stellen der Übergang zwischen Felge und Reifen nicht sauber eingezeichnet war.

Für die Fälle einer Ellipse, die sich deutlich vom Hintergrund abzeichnet, funktioniert die Ellipsensuche somit schon in der nicht optimierten Variante gut.



Abbildung 52: Testergebnis einer dunklen Felge auf dunklem Reifen. Das Kantenbild rechts zeigt die Erkennungsschwierigkeit aufgrund vieler starker Kanten durch einen geriffelten Übergang zwischen Felge und Reifen.

Bei dunklen Felgen auf dunklen Reifen, wie in Abb.52 zu sehen, ist die Erkennung der Ellipsen sehr schwierig, da die geriffelte schwarze Felge durch die Reflektion der Beleuchtung an unregelmäßigen Stellen starke Kanten hervorruft, die mehrere Möglichkeiten einer Ellipsenbildung zulassen. Zur Erkennung dieser Sorte von Felgen muss eine Anpassung des Algorithmusses gefunden werden, der die richtige Kombination der einzelnen Kantenstücke zusammensetzt, die im Kantenbild zu erkennen sind.

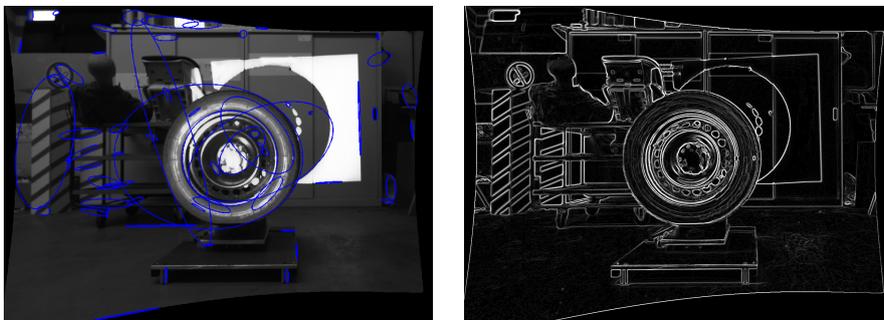


Abbildung 53: Testergebnis der dunklen Felge bei großer Entfernung zur Kamera

Versetzt man das Testrad um 40 Zentimeter nach hinten, wie in Abb.53 zu erkennen, wird die Felge augenscheinlich richtig erkannt. Dies stellt jedoch keine geeignete Lösung für das Suchproblem auf eine dunklen Felge auf dunklem Reifen dar, da die Ellipse so nur wenige Pixel umfasst, und damit der Genauigkeitsanforderung nicht gerecht werden kann. Das Testrad sollte somit in näherer Entfernung zur Kamera stehen.

Eine weitere Möglichkeit der relativ gesehen guten Erkennungsrate dieser Testbilder kann die helle Beleuchtung sein, die die Reflexionen der Felge so stark werden lässt, dass sie gemeinsam eine Kante bilden, statt viele einzelne Kanten für kleine Reflexionen. Da dasselbe Rad noch einmal mit dunklerer Beleuchtung getestet

wurde und dabei eine deutliche schlechtere Erkennungsrate zeigt, wird dies in einem späteren Test (siehe Kapitel 5.5) evaluiert werden müssen.

Felgen in Bildern mit Strukturlicht, welches häufig für Stereomessungen verwendet wird (siehe Abb.54), können nicht erkannt werden, da auf Grund der Lichtverhältnisse keine Kanten der Felgen, sondern nur die des Strukturlichtes erkannt werden. Von der Benutzung von Strukturlicht kann somit nur abgeraten werden.

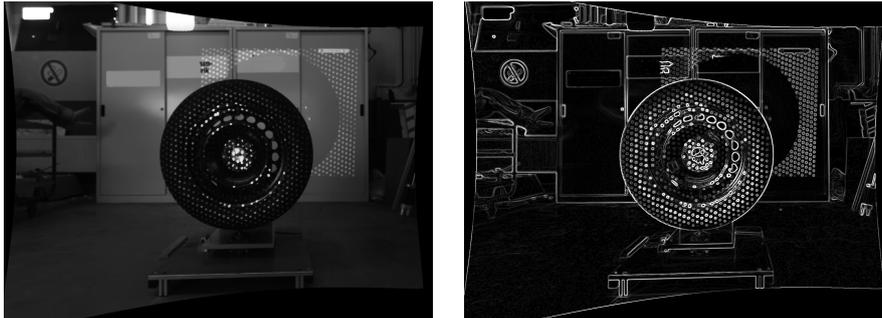


Abbildung 54: Strukturbeleuchtung auf ein Rad und die Auswirkung auf die Kantenerkennung

Sechs Bilder eines Fahrzeuges zeigen das nach den bisherigen Radtests erwartete Ergebnis einer Suche auf hellen Felgen, dass die helle Felge bei den meisten (4 von 6) Bildern erkannt wurde. Bei den 2 übrigen Bildern war die Beleuchtung hierfür zu schwach, sodass nicht die gesamte Kante der Felge erkannt wurde. Zusätzlich zur Felge wurde auch der Radkasten korrekt dargestellt. Siehe hierzu Abb.55.



Abbildung 55: Fahrzeug mit erkannter Felge und Radkasten

5.3.1 Fazit der Evaluation an Radbildern

Aus den Tests lassen sich einige Informationen für die weitere Bearbeitung ableiten:

- Strukturbeleuchtung
Strukturbeleuchtung ist zwar für die Stereoskopie sehr gut, nicht aber für die Ellipsendetektion, da durch diese Beleuchtung starke Kanten an falschen Stellen auf dem Rad entstehen, die das Ergebnis der Ellipsendetektion so

verfälschen, dass sie auf ein falsches Ergebnis liefert. Es ist somit zu vermeiden, die Ellipsendetektion auf strukturbeleuchteten Rädern anzuwenden.

- **Schwarze Felgen auf schwarzen Reifen**
Schwarze Felgen heben sich von schwarzen Reifen nur durch sehr schwache Kanten ab. Reflexionen wirken sich daher stark auf die Ellipsendetektion aus, da sie in diesem Fall stärkere Kanten auf der Felge erzeugen, als der gesuchte Übergang. Da sich Reflektionen durch die unterschiedliche Kamerastellung bei der Stereoaufnahme an verschiedenen Stellen auf den beiden entstandenen Bildern auswirken, wird hier die Ellipsendetektion erheblich erschwert. In einem weiteren Test muss somit beobachtet werden, ob sich unterschiedliche Belichtungen positiv auf die Detektion dieser Räder auswirkt.
- **Kantenbilder mit unterbrochenen Linien**
Bei schwachen Kanten kommt es vor, dass diese an einigen Stellen so schwach sind, dass sie durch den Schwellwert durchfallen, der die Kanten vom Hintergrund trennt. Dabei entstehen Lücken in den Kanten und sie können durch das Labeling nicht mehr zu einer einzelnen Kontur zusammengesetzt werden, auf der die Ellipsensuche durchgeführt wird. Hierdurch werden einige Ellipsen nicht korrekt erkannt, da die dafür nötigen 60% der Ellipse nicht vorhanden sind. Bei der Optimierung auf den Anwendungsfall muss daher ein Algorithmus implementiert werden, der dieses Problem löst.

5.4 Evaluation des Verfahrens zur Labelzusammensetzung unterbrochener Ellipsen

5.4.1 Test an künstlichen Ellipsen

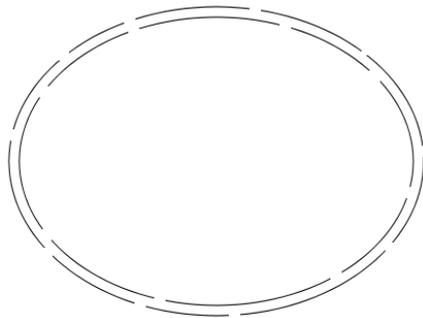


Abbildung 56: Ellipsen mit Unterbrechungen

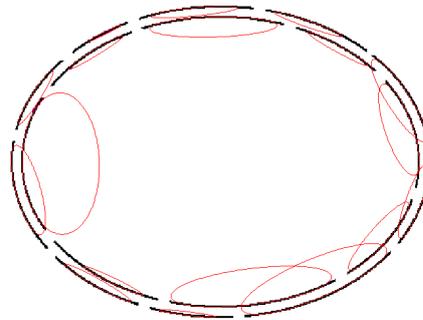


Abbildung 57: Ergebnis des Fittings mit Label

Beschreibung

Mit diesem Test wird die Funktion der Zusammensetzung von Ellipsenteilen zu einer ganzen Ellipse getestet. Als Ausgangsbild dienen die unterbrochenen Ellipsen aus Abb.56, bei denen das Ergebnis eines einfachen Fittings mit vorherigem Labeln des Bildes das Ergebnis aus Abb.57 ergibt. Die Algorithmen aus Kapitel 4.3.1 werden auf die Ellipsenteile angewandt, wobei das Ziel darin besteht, diese zu korrekten Ellipsen zusammen zu setzen.

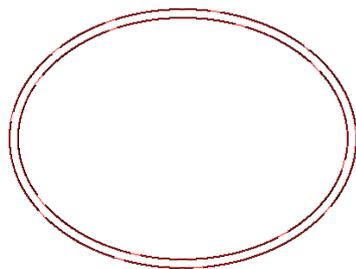


Abbildung 58: Ergebnis ohne Labeling

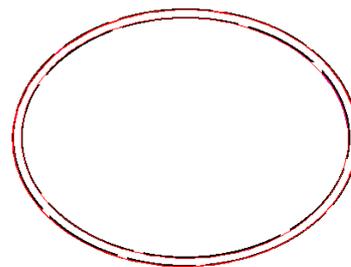


Abbildung 59: Ergebnis mit Verbindung der Label

Ergebnis

Das Ergebnis des Experimentes ist in den Abbildungen Abb.58 und Abb.59 zu erkennen. Sowohl der Algorithmus, bei dem die Label zusammengefügt werden, als auch das Fitting ohne vorheriges Labeln haben die Ellipsen richtig zusammengefügt. Gezeigt wurde somit, dass sich beide Verfahren in der Theorie zum

Zusammensetzen der Ellipsenteile eignen.

5.4.2 Test an Radbildern

Beschreibung

Die Eignung für die Praxis wird in einem weiteren Test untersucht, bei dem auf zwei verschiedene Felgen mit unterschiedlichen Beleuchtungsbedingungen beide Algorithmen angewandt werden. Das Ergebnis dieses Tests ist in Tabelle 7 aufgeführt. Jede Beleuchtungseinstellung wurde dabei fünf mal aufgenommen, um die Wiederholgenauigkeit zu berücksichtigen.

Deutung

Auffällig ist bei diesem Testergebnis dass das Verbinden der Label in keinem Fall das Ergebnis verbessert, es dafür aber in einigen Fällen verschlechtert. Dies kann damit erklärt werden, dass bei einer falschen Auswahl der zu verbindenden Label das Ergebnis möglicherweise besser als vorher ist. Da aber keine Möglichkeit implementiert ist die Label bei einer möglicherweise auftretenden noch besseren Kombination wieder zu trennen, kann diese nicht verwendet werden.

Mit einer einfachen Verbindungstechnik ist das Problem der aufgetrennten Label somit nicht zu lösen, weitere Versuche werden in dieser Richtung nicht übernommen, da dieses Forschungsgebiet der Strukturzusammensetzung genügend Thematik für eine eigenständige Diplomarbeit bietet.

Nr	Felgentyp	Beleuchtung	o-lk	o-rk	m-lk	m-rk
0	schwarze Felge	Blooming	-	-	-	-
1	schwarze Felge	Blooming	-	-	-	-
2	schwarze Felge	Blooming	-	-	-	-
3	schwarze Felge	Blooming	-	-	-	-
4	schwarze Felge	Blooming	-	-	-	-
5	schwarze Felge	kurz vor Blooming	-	-	-	-
6	schwarze Felge	kurz vor Blooming	-	-	-	-
7	schwarze Felge	kurz vor Blooming	-	-	-	-
8	schwarze Felge	kurz vor Blooming	-	-	-	-
9	schwarze Felge	kurz vor Blooming	-	-	-	-
10	schwarze Felge	keine Beleuchtung	-	-	-	-
11	schwarze Felge	keine Beleuchtung	-	-	-	-
12	schwarze Felge	keine Beleuchtung	-	-	-	-
13	schwarze Felge	keine Beleuchtung	-	-	-	-
14	schwarze Felge	keine Beleuchtung	-	-	-	-
15	große Silberfelge	keine Beleuchtung	x	x	x	x
16	große Silberfelge	keine Beleuchtung	x	x	x	x
17	große Silberfelge	keine Beleuchtung	x	x	x	x
18	große Silberfelge	keine Beleuchtung	x	x	x	x
19	große Silberfelge	keine Beleuchtung	x	x	x	x
20	große Silberfelge	etwas blooming	x	x	x	x
21	große Silberfelge	etwas blooming	x	x	x	x
22	große Silberfelge	etwas blooming	x	x	x	x
23	große Silberfelge	etwas blooming	x	x	x	x
24	große Silberfelge	etwas blooming	x	x	x	-
25	große Silberfelge	Blooming	x	x	-	x
26	große Silberfelge	Blooming	x	x	x	-
27	große Silberfelge	Blooming	x	x	x	x
28	große Silberfelge	Blooming	x	x	x	x
29	große Silberfelge	Blooming	-	-	-	-

Tabelle 7: Vergleichsergebnis der Ellipsenerkennung ohne und mit Verbinden der Label

o-lk und o-rk: linke und rechte Kamera ohne Verbinden der Label

m-lk und m-rk: linke und rechte Kamera mit Verbinden der Label

ein x steht für eine erkannte Ellipse, bei einem - wurde keine korrekte Ellipse gefunden

5.5 Test an einzelnen Rädern

Beschreibung

Zur Evaluation der Erkennungsqualität der Einzelraderkennung wurde der Erkennungsalgorithmus ohne Binarisierung im Vergleich mit den zwei beschriebenen Binarisierungsvarianten aus Kapitel 4.3.2 auf fünf verschiedene Felgentypen unter unterschiedlichen Beleuchtungssituationen angewandt. Jede Einstellung wurde dabei 15x aufgenommen, um die Wiederholgenauigkeit feststellen zu können. Insgesamt gibt es drei verschiedene Beleuchtungen:

- keine Beleuchtung
Hierbei wird außer dem vorhandenen Umgebungslicht keine zusätzliche Beleuchtung verwendet, sondern der Gain und die Shutterzeit bei den Kameras so hoch gedreht dass die Felgen für ein menschliches Auge gut erkennbar sind, wodurch das Bildrauschen sehr stark wird.
- etwas blooming
Aufgrund der metallenen Oberfläche mit ihren guten Reflexionseigenschaften, vor allem bei den hellen Felgen, tritt blooming auf, sobald eine zusätzliche Beleuchtung genutzt wird. Die hier verwendete Einstellung ist so gewählt, dass nur sehr leichtes Blooming auftritt, bei guter Erkennbarkeit der Felgen.
- Überstrahlt
Wie in Kapitel 5.3.1 erwähnt werden in dieser Beleuchtungsvariante die dunklen Felgen überstrahlt um zu testen, ob diese Einstellung die Felgen besser hervorstehen lässt als die Reifen.



Abbildung 60: Verwendete helle Felgen des 2D Tests
von links nach rechts: große Silberfelge, gewölbte Silberfelge, kleine Silberfelge

Die Felgentypen beinhalten drei helle Felgen, die in Abb.60 dargestellt sind. Durch die unterschiedlichen Felgenwölbungen und die damit verbundene unterschiedliche Lichtreflexion stellt jede dieser Felgen ein anderes Erkennungsszenario dar.

Die zwei dunklen Felgen des Tests sind in Abb.61 abgebildet und bestehen aus einer schwarzen Felge mit geriffeltem Übergang zwischen Felge und Reifen, sowie einer rostigen Felge die einen verbeulten Übergang zwischen Felge und Reifen bildet.

Ergebnis

Das Ergebnis des Tests ist in Tabelle 8 aufgelistet. Die teils großen Erkennungsunterschiede zwischen linker und rechter Kamera bei fehlender Zusatzbeleuchtung

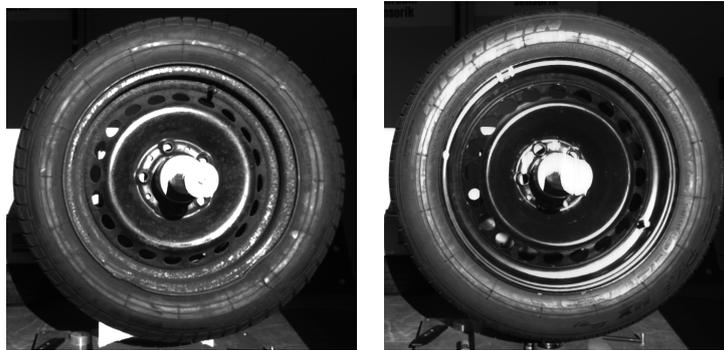


Abbildung 61: Verwendete dunkle Felgen des 2D Tests
links: rostige Felge, rechts: schwarze Felge

resultieren aus den unterschiedlichen Helligkeitsbedingungen, da beim Versuchsstand das Tageslicht von zwei Fensterreihen kam, durch die der Reifen für die linke Kamera stärker beleuchtet wurde als für die rechte Kamera. Auch durch Drehung des Kamerasystems ließ sich dieser Effekt nicht verhindern, sondern nur auswählen, welche der Kameras stärker beleuchtet wurde. Da dieser Effekt bei fehlender Extrabeleuchtung auch im Produktivfeld aufträte ist er allerdings realistisch und der Test kann so gewertet werden.

Die dunklen Felgen werden mit keinem der verwendeten Verfahren erkannt. Bei

Nr	Felgentyp	Beleuchtung	Original		Otsu1*		Otsu2**	
			l	r	l	r	l	r
0	große Silberfelge	etwas Blooming	15	14	15	14	11	15
15	große Silberfelge	keine	15	15	11	2	15	15
30	schwarze Felge	keine	0	0	0	0	0	0
45	schwarze Felge	etwas blooming	0	0	0	0	0	0
60	schwarze Felge	überstrahlt	0	0	0	0	0	0
75	dunkle Rostfelge	überstrahlt	0	0	0	0	0	0
90	dunkle Rostfelge	etwas blooming	0	0	0	0	0	0
105	dunkle Rostfelge	keine	0	0	0	0	0	0
120	gewölbte Silberfelge	keine	8	9	15	15	13	15
135	gewölbte Silberfelge	etwas blooming	0	0	0	0	0	0
150	kleine Silberfelge	etwas blooming	15	15	15	15	15	15
165	kleine Silberfelge	keine	2	4	12	0	15	15

Tabelle 8: Ergebnis der 2D Raddetektion

* Otsu1: Binarisierung vor Kantendetektion

** Otsu2: Binarisierung nach Medianfilter

dunkler Beleuchtung sind die Kanten zwischen Felge und Reifen zu schwach um lückenlos erkannt zu werden. Bei heller Beleuchtung sind die Kanten durch Reflexionen unterbrochen und bilden keine zusammenhängende Linie mehr. Zudem werden bei einigen dunklen Felgen bei heller Beleuchtung sämtliche Linien auf der Felge deutlicher angezeigt, als der Übergang zwischen Felge und Reifen (siehe

Abb.62). Da diese Linien aber auch durch Reflexionen unterbrochen sind, können sie vom Verfahren nicht richtig zugeordnet werden.

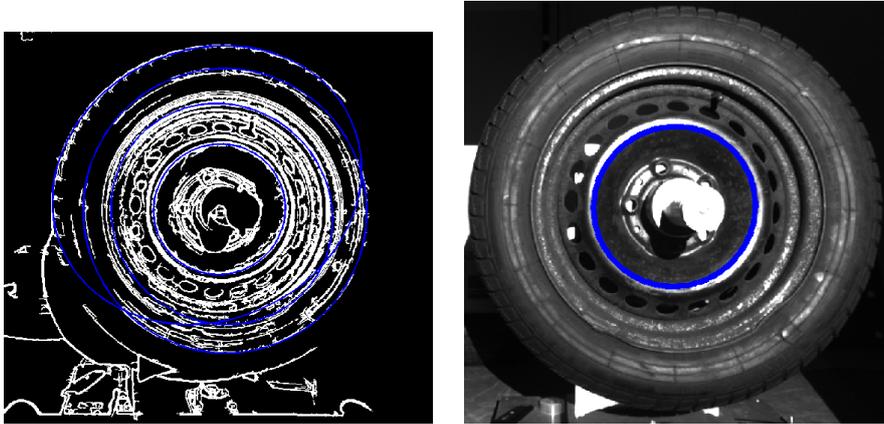


Abbildung 62: Bei dunklen Felgen sind bei heller Beleuchtung so viele Felgenteile im Kantenbild zu erkennen, dass das Verfahren keine Ellipsenteile richtig zuordnen kann

Der erhoffte Unterscheidungseffekt durch Überstrahlung des Bildes (siehe Kapitel 5.3.1) blieb aus, da so auch die Reifen Reflexionen bilden, die das Ergebnis verfälschen.

Aus diesem Test heraus lässt sich somit sagen, dass dunkle Felgen mit den hier verwendeten Verfahren nicht erkannt werden können.

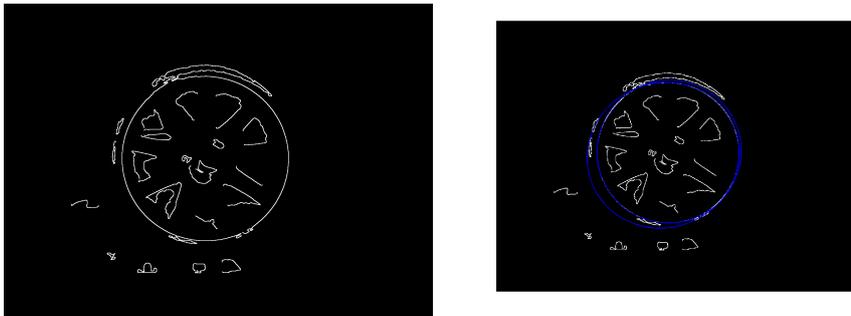


Abbildung 63: Verfälschung des Kantenbildes durch Blooming und Reflexionen auf den Reifen

Bei den hellen Felgen erreicht das Verfahren Otsu2 die beste Erkennungsleistung. Durch die Binarisierung des Ausgangsbildes nach der Kontrastoptimierung wird sowohl eine zusätzliche Robustheit gegen Helligkeitsschwankungen erreicht, als auch die Vermeidung von konzentrischen Ellipsen an den Übergängen zwischen Felge und Reifen. Dies zeigt sich besonders in der Erkennung der kleinen Silberfelge, bei dem nur dieses Verfahren ein zufriedenstellendes Ergebnis erreicht. Schwächen zeigt es jedoch bei zusätzlicher Beleuchtung und dadurch entstehendem Blooming,

da dieses den Übergang zwischen Felge und Reifen an einigen Stellen verwischt und somit zu falschen Objektverknüpfungen im Kantenbild führen kann, wie im Beispiel von Abb.63 dargestellt, bei dem die Kante der Felge bis in den Reifen führt. Da dieses Verhalten nur selten auftritt, wird in den weiteren Schritten die Bildvorverarbeitungsvariante Otsu2 anstelle des Originalalgorithmusses benutzt.

5.6 Test am vorbeifahrenden Fahrzeug

Der Gesamtalgorithmus zur Verfolgung von Rädern wird zuerst an künstlich mit 3D Studio Max erzeugten vorbeifahrenden Kreisen getestet und anschließend auf Bildsequenzen von an den Kameras vorbeifahrenden Autos angewandt. Da der Algorithmus bei der Evaluation an Einzelradbildern in Kapitel 5.5 schon gezeigt hat, dass er keine dunklen Felgen erkennen kann, wird für die Vorbeifahrt ein Auto mit hellen Felgen genutzt.

5.6.1 Verfolgung künstlicher 3D Kreise

Zur Evaluation des Algorithmusses wurde mit 3d Studio Max Version 8 ein 3D-Testkreis erzeugt (siehe Abb.64), der in einem Abstand von 4 Metern an zwei virtuellen Kameras mit bekannter Position vorbeifährt.

Als Ergebnisse des Tests werden Graphen ausgegeben, die anzeigen, auf welcher Position der Algorithmus das Rad im jeweiligen Testbild erwartet hat und auf welcher Position es tatsächlich war, sowie diejenigen Bildnummern, bei denen der Trackingalgorithmus auf Grund guter Vorhersagen die Berechnung auf Bildausschnitte beschränken konnte, sowie die Bilder bei denen sich der Algorithmus unsicher ist, ob er die richtigen Ellipsen gefunden hat.



Abbildung 64: Der für den Test der Vorbeifahrt verwendete Testkreis aus linker und rechter Kamera

Getestet wurde eine Vorbeifahrt des Kreises an der Kamera, sowie eine Drehung des Rades um 180 Grad, da die Vorbeifahrt und Drehung eines Rades die für die Anwendung wichtigen Abläufe darstellen.

Ergebnis der Vorbeifahrt

Die Bilder der Vorbeifahrt wurden erwartungsgemäß erkannt. Sobald die ersten 3 Bilder, bei denen in beiden Kameras die Ellipsen zu erkennen sind, auftauchen, verkleinert der Algorithmus die Suchgröße auf den Bildausschnitt, in dem die nächste Ellipse erwartet wird. Sämtliche anderen Bilder, bei denen eine der beiden Ellipsen der Kreisprojektion nicht korrekt erkannt werden kann, werden als möglicher Messfehler markiert. In den Graphen der Hauptachsenlänge und der X-Position des Mittelpunktes der Ellipsen in der linken Kamera in Abb.65 ist zu erkennen, dass die Vorhersage (Rot) sich in den ersten paar gemeinsamen Bildern dem tatsächlichen Wert anpasst. Da sie erst dann genutzt wird, wenn sich der Algorithmus ob ihrer Korrektheit sicher ist, werden die darauf folgenden gemeinsamen Ellipsen gut

erkannt.

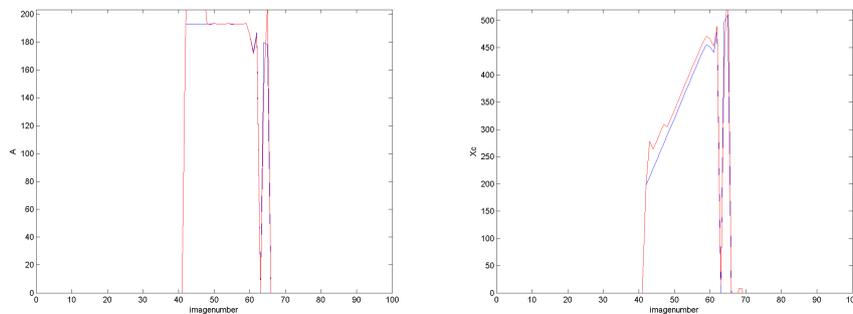


Abbildung 65: Graph der Hauptachsen- und X-Mittelpunktverfolgung der linken Kamera bei der Vorbeifahrt des Testkreises

Ergebnis der Drehung

Die Ergebnisse der Drehung sind so gut wie die der Vorbeifahrt. Nachdem das Vorwissen darauf angepasst wurde, dass der Reifen nicht mehr gerade ins Bild läuft, sondern um 90° gedreht im Bild erscheint, wird die Reifengröße gut erkannt, was sich auch gut im Verlauf der horizontalen Achse in Abb.66 ablesen lässt. Die Vorhersage ist jedoch wie bei der Drehung bei manchen Punkten etwas vorläufig, schätzt also die Bewegung teilweise zu schnell ein. Dies hat jedoch keinerlei negative Auswirkungen auf die Erkennung.

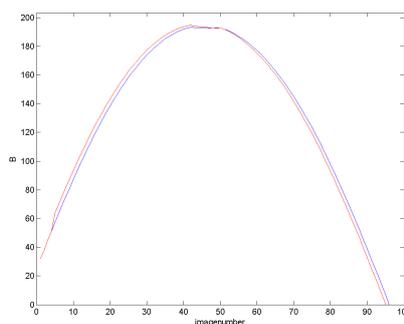


Abbildung 66: Graph der horizontalen Achse bei der Drehung des Testkreises

Deutung

Wie erwartet funktioniert die Verfolgung eines vorbeifahrenden künstlich erzeugten Kreises sehr gut, wie auch die einer Drehung des Kreises. Das Tracking verkürzt dabei die Suchzeit ab einer gewissen Menge gefundener Ellipsen, ansonsten ist die Verfolgung eine Erkennung auf Einzelbildern.

5.6.2 Verfolgung von Autorädern in Stereobildern

Der komplette Algorithmus wird nun auf Autoräder angewandt. Hierzu wurden 6 Vorbeifahrten eines Autos, sowie ein kompletter Lenkeinschlag mit dem in Kapitel 2.6 vorgestellten Kamerasystem aufgenommen. Da der Erkennungsalgorithmus bei der Verfolgung von Autorädern derselbe wie bei der Einzelraderkennung ist, wird erwartet, dass alle Felgen erkannt werden. In Abb.67 sind 2 Beispielbilder zur besseren Vorstellung der Aufnahmen zu sehen.



Abbildung 67: Beispielbilder der verwendeten Aufnahmen. Links das eingeschlagene Rad zur Erkennung eines Lenkeinschlags, rechts eine Aufnahme des vorbeifahrenden Autos

Ergebnis

Die Abb.68 zeigt das Ergebnis einer der Vorbeifahrten eines Rades. Es ist zu erkennen, wie das Rad zwischen dem 20. und 30. Bild soweit im Kamerabild erscheint, dass es erkannt werden kann und dann ab dem 36. Bild sicher erkannt wird.

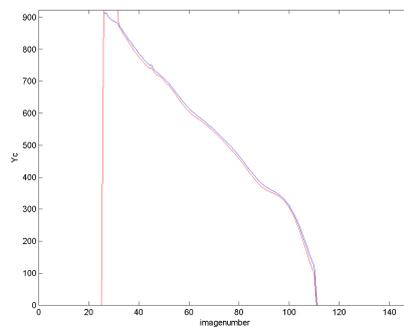


Abbildung 68: Verschiebung des Rades bei einer Vorbeifahrt
Der rote Graph zeigt die geschätzte Position, der blaue Graph die tatsächliche

Eine Vorbeifahrt des Autos, bei dem beide Räder einer Seite nacheinander an der Kamera vorbeifahren ist in Abb.69 als Graph der X-Koordinate des Rades zu sehen. Hierbei fährt das Auto mit dem ersten Rad an der Kamera vorbei und hält so an, dass das zweite Rad im Kamerabild verweilt. Daher ist bei dem zweiten Rad ab dem 100. Bild keine Veränderung mehr zu erkennen.

Das Ergebnis des Lenkeinschlages ist in Abb.70 zu sehen, das die Veränderung der Hauptachse des Rades zeigt. Beim Aufnahmebeginn ist das Rad eingeschlagen,

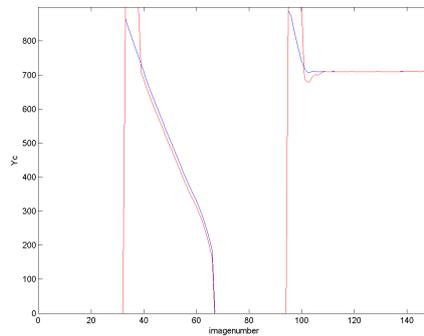


Abbildung 69: Verschiebung des Rades bei einer Vorbeifahrt
Der rote Graph zeigt die geschätzte Position, der blaue Graph die tatsächliche

bis zur Aufnahmemitte wird das Rad einmal komplett in die andere Richtung eingeschlagen, danach wieder zurück. Beim Einschlag in der Bildmitte wird das Rad dabei von der Lichtquelle weggedreht, wodurch einige Aufnahmeschwierigkeiten entstehen, die jedoch vom Algorithmus erkannt und notiert werden.

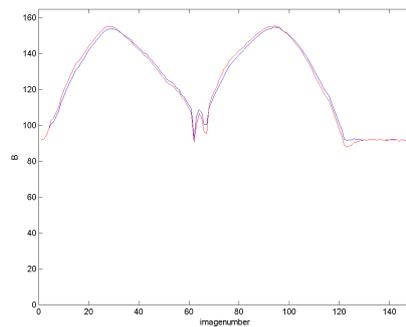


Abbildung 70: Lenkeinschlag eines Rades
Der rote Graph zeigt die geschätzte Position, der blaue Graph die tatsächliche

6 Zusammenfassung

In dieser Diplomarbeit wurden die Möglichkeit evaluiert, mit einem Ellipsenerkennungsverfahren pixelgenau Autoräder zu erkennen. Dazu wurden zwei Ellipsenverfahren ausgewählt und auf ihre Eignung hinsichtlich der Aufgabe überprüft. Als für die Aufgabe am besten geeignetes Verfahren hat sich das Ellipsenfitting mit RANSAC erwiesen, welches in weiteren Tests auf die optimalen Parametereinstellungen hinsichtlich der Aufgabenstellung getestet wurde.

Um Bilder von vorbeifahrenden Autos und Autorädern zu bearbeiten, wurde ein neuartiges Filterverfahren für die Kantenbilder entwickelt. Dieses nutzt die Gradientenhistogramme von Objekten, die aus zusammenhängenden Pixeln erstellt wurden, um diejenigen Elemente mit zu geringer Krümmung zu entfernen.

Die auf den Stereobildern gefundenen Ellipsen wurden miteinander verglichen und die am besten zueinander passenden im Bild belassen. Um die besten Paare zu finden wurden dazu deren Höhe und senkrechte Achse miteinander verglichen, da aus dem Vorwissen der Stereoaufnahmen bekannt war, dass sich diese Parameter gleichen müssen. Abschließend wurde gezeigt, dass aus den gewonnenen Ellipsen leicht 3D Informationen rekonstruiert werden können.

6.1 Fazit

Die pixelgenaue Raderkennung mittels Ellipsensuche in den Stereobildern hat sich leider aus verschiedenen Gründen als für dunkle Felgen nicht durchführbar erwiesen. Zum einen sind viele Felgen beim Übergang zwischen der Felge zum Reifen mit einer Kante versehen, die im Kantenbild als eng nebeneinander liegende parallele Ellipsen erscheint (siehe z.B. Abb.52 für dunkle Felgen und Abb.41 für helle Felgen). Während es bei hellen Felgen möglich ist, diesem Effekt durch eine vorherige Binarisierung des Ausgangsbildes vorzubeugen, ist dieses Verfahren bei dunklen Felgen nicht einsetzbar, da sich die dunklen Felgen nicht deutlich von den dunklen Reifen abheben und somit beide Elemente denselben Binärwert erhielten.

Desweiteren heben sich die dunklen Felgen in einigen Fällen, besonders bei der rostigen Felge, farblich kaum vom Reifen ab. Abb.71 zeigt die Kantenbilder der dunklen Rostfelge bei verschiedenen Beleuchtungsstärken. Wie man hier deutlich erkennen kann, ist der Übergang zwischen Felge und Reifen kaum von den Kanten der Reifemuster, bzw. der Felge selbst unterscheidbar.

Bei den metallenen Oberflächen der Felgen entstehen zudem schon bei kleiner Beleuchtung Reflexionen auf der Felge. Da sich die Kanten der Reflexionen mit denen des Felgenmusters im Kantenbild überlagern, werden die Kanten auf der Felge vom Labeling als zusammenhängendes Objekt gewertet. Durch diesen Effekt werden nicht mehrere Ellipsen auf dem Felgenmuster gefunden, sondern nur eine Einzelne. Trennt man die Objekte auf, erhält man jedoch viele einzelne Teile, die sich nicht durch einen einfache Algorithmus zusammensetzen lassen, da sich dicht beeinander liegen und eine ähnliche Form haben. Auch auf Vorwissen über die Form kann man nicht zurückgreifen, da die Ellipse durch verschiedene Achsverhältnisse und Größen sehr viele Formen annehmen kann, von denen die Gesuchte nicht bekannt ist. Dadurch werden häufig Ellipsen gefunden, die aus falschen Ellipsenteilen parallel zueinander liegender Ellipsen bestehen (siehe Kapitel 5.4.1).

All diese Probleme treten bei hellen Felgen nicht auf, wenn diese vorher binarisiert werden. Da dieses Verfahren sehr gute Ergebnisse im Test gezeigt hat, kann somit

abschließend gesagt werden, dass es mit der gewählten Methode nicht möglich ist, Räder mit dunklen Felgen pixelgenau zu detektieren. Verfügt das Rad über eine helle Felge, kann diese mit einer sehr hohen Wahrscheinlichkeit pixelgenau gefunden werden.

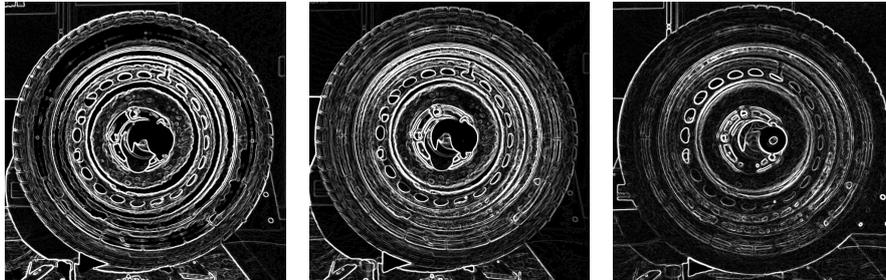


Abbildung 71: Kantenbilder der dunklen Rostfelge bei verschiedenen Beleuchtungsstärken

6.2 Ausblick

Da sich diese Form der Raderkennung nur für die Suche nach hellen Felgen eignet, ist sie so im produktiven Umfeld nicht einsetzbar. Es sind jedoch Erweiterungen möglich, mit denen die schlechte Erkennungsqualität verbessert werden könnte. Da diese Erweiterungen den Rahmen dieser Diplomarbeit überschreiten würden, wurden sie hier nicht mehr weiter verfolgt.

Eine Möglichkeit besteht in der Segmentierung des Ausgangsbildes. Da der Reifen eine relativ konstante Farbe besitzt, könnte der Übergang zur Felge durch den Farbübergang vom Reifen zur Felge gefunden werden. Da Reflexionen mit dieser Methode als eigenständiges Objekt klassifiziert werden können, kann man sie so möglicherweise aus dem Bild entfernen. Eine weitere Möglichkeit besteht in einer besseren Objektzusammensetzung der Kanten.

Da alle Erweiterungen jedoch zusätzliche Berechnungen benötigen ist ein komplett anderer Ansatz für einen Einsatz im produktiven Umfeld empfehlenswerter. Denkbar wäre hier z.B. eine Laservermessung des Reifens, bei der Reflexionen und Farben nicht die starken negativen Auswirkungen haben wie bei der optischen Erkennung. Auch ein Infrarotsensor, der die unterschiedliche Wärmereflexion von Felge und Reifen misst ist denkbar, wobei es auch hier Schwierigkeiten bei Radkappen aus Plastik geben könnte.

7 Anhang

7.1 Anteil eigener und fremder Verfahren

Bei vielen der hier gezeigten Verfahren kommen Algorithmen vor, die es zu einem großen Teil schon vor dieser Arbeit gab und daher nicht noch einmal implementiert wurden. Die folgende Liste gibt Auskunft darüber, welche Algorithmen von welcher Quelle stammen.

Nutzung fremder Verfahrensimplementierungen

- Fittingverfahren aus [HF98]
- Programmgerüst aus [Inv02]
- Eckendetektor aus [HY04]
- Otsuimplementierung von <http://emfs1.eps.hw.ac.uk/~ceeyrp/WWW/Teaching/B39SD2/Tutorials/Otsu.m>
- Strukturtensor von Andreas Haja zur Gewinnung der Kanten und Bestimmung der Gradienteninformationen eines Bildes
- Non-Maximum-Suppression (**Kommentar: Verweis auf Kovesi Toolbox**)

Nutzung von Funktionen aus Matlab

- `adapthisteq` für contrast limited adaptive histogram equalization
- `medfilt2` für den Medianfilter
- `BWLabel` für das Labeln eines schwarz-weiss Bildes
- `bwmorph` für Closing und Dilatation

Eigene Implementation beschriebener Verfahren

- RANSAC Teil des Fittingverfahrens
- Erweiterung der Houghtransformation zur RRHT
- Tangentensuche der RRHT
- Berechnung der 3D-Koordinaten

Eigenentwicklungen von Verfahren

- Löschen störender Objekte durch Nutzung eines Gradientenhistogramms
- Abweichungsmessung von Tangenten
- Suche der richtigen Ellipsenkombinationen aus Stereobildern
- Zusammensetzen unterbrochener Konturen
- Trackingverfahren

Abbildungsverzeichnis

1	Messergebnisses für Fitting	8
2	Invarianz gegenüber euklidischer Transformation	9
3	Unterschied euklidischer zu algebraischem Abstand	10
4	Wirkung von RANSAC auf Fitting	10
5	Ablaufplan RANSAC	11
6	Darstellung des Houghraums	12
7	Schwellwerterkennung mit Otsu	13
8	Schwachstelle des Otsuverfahrens	14
9	Histogrammebnung	15
10	Beispielbild CLAHE	15
11	Vergleich von Median und Averagefilter	16
12	Harris Kantendetektor	18
13	Eckendetektion	18
14	Graddfirealgorithmus	19
15	Labeling	19
16	Erosion	20
17	Dilatation	20
18	Der verwendete Messstand	22
19	Blooming	23
20	Kamerarauschen	23
21	schematische Darstellung des Stereonormalfalls Quelle: eigene Nachbildung einer Darstellung aus [Luh00] Seite 310	24
22	Bestimmung des Ellipsenzentrum bei der Houghtransformation	26
23	allgemeiner Kegelschnitt	27
24	gelabelte Ellipsen	27
25	Ellipsenregionen	27
26	Ausreißer beim Fitting	30
27	Ergebnis einer Kantenerkennung	31
28	2 Ellipsen in einer Kontur	32
29	Linie mit Kurvenende	33
30	Histogramm der Linie mit Kurvenende	34
31	Zwischenspeicher der Kantenfilterung	34
32	Gradientenhistogramm der links dargestellten schmalen Ellipse	35
33	Ergebnis der Kantenfilterung	35
34	Vergleich der Tangentenbestimmungsverfahren	39
35	Schaubild der RANSAC Nutzung zur Ellipsensuche	41
36	Ellipsensuche auf nah beeinander liegenden Ellipsen	41
37	Ellipsensuche auf einer unterteilten Ellipse	42
38	Ablaufpläne der Verfahren zur Ellipsendetektion	43
39	Beispiel einer Störung, die die Konturzusammensetzung erschwert	44
40	Ablaufplan der Objektverbindung	45
41	Ergebnis der Binarisierung	47
42	Ellipsen des linken und rechten Rades vor dem Vergleich	47
43	Ellipsen des linken und rechten Rades nach dem Vergleich	48
44	Ablaufplan der Einzelraderkennung	48
45	Ablaufplan der Radverfolgung	49
46	künstliche Testellipse mit Rauschen	51
47	Winkelfehler der Verfahren zur Tangentenbestimmung	52

48	Vergleich der Ellipsenverfahren	53
49	Wahrscheinlichkeitsgraph richtiger Ergebnisse	54
50	Zur Berechnung nötiger Ellipsenanteil	57
51	Testergebnis einer hellen Felge auf dunklem Reifen	59
52	Testergebnis einer dunklen Felge auf dunklem Reifen	60
53	Testergebnis der dunklen Felge bei großer Entfernung zur Kamera	60
54	Auswirkung von Strukturbeleuchtung	61
55	Testergebnis eines Fahrzeuges	61
56	Ellipsen mit Unterbrechungen	63
57	Ergebnis des Fittings mit Label	63
58	Ergebnis ohne Labeling	63
59	Ergebnis mit Verbindung der Label	63
60	Verwendete helle Felgen des 2D Tests	66
61	Verwendete dunkle Felgen des 2D Tests	67
62	Erkennungsschwierigkeit bei dunklen Reifen	68
63	Verfälschung des Kantenbildes durch Blooming und Reflektionen auf den Reifen	68
64	Der für den Test der Vorbeifahrt verwendete Testkreis aus linker und rechter Kamera	70
65	Graph der Hauptachsen- und X-Mittelpunktverfolgung der linken Kamera bei der Vorbeifahrt des Testkreises	71
66	Graph der horizontalen Achse bei der Drehung des Testkreises	71
67	Beispielbilder des Gesamttests	72
68	Verfolgung eines Einzelrades	72
69	Verfolgung eines Einzelrades	73
70	Drehung eines Einzelrades	73
71	Kantenbilder der dunklen Rostfelge bei verschiedenen Beleuchtungsstärken	75

Literatur

- [AT04a] ACHLER, O. ; TRIVEDI, M.M.: Camera based vehicle detection, tracking, and wheel baseline estimation approach. In: *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*, 2004, S. 743–748
- [AT04b] ACHLER, O. ; TRIVEDI, M.M.: Vehicle wheel detector using 2D filter banks. In: *Intelligent Vehicles Symposium, 2004 IEEE*, 2004, S. 25–30
- [Boo79] BOOKSTEIN, F.: Fitting conic sections to scattered data. In: *Computer Graphics and Image Processing* (1979), S. 9:56–71
- [Can86] CANNY, J: A computational approach to edge detection. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (1986), Nr. 6, S. 679–698. – ISSN 0162–8828
- [CL04] CHENG, Z. ; LIU, Y.: Efficient technique for ellipse detection using restricted randomized Hough transform. In: *Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004. International Conference on Bd. 2*, 2004, S. 714–718 Vol.2
- [CYW04] CAI, Wenchao ; YU, Qian ; WANG, Hong: A fast contour-based approach to circle and ellipse detection. In: *Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on Bd. 5*, 2004, S. 4686–4690 Vol.5
- [Dav96] DAVIES, E.R.: *Machine Vision*. 2nd. Academic Press, 1996, 1996
- [FB81] FISCHLER, Martin A. ; BOLLES, Robert C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. In: *Commun. ACM* 24 (1981), S. 381–395
- [FF95] FITZGIBBON, A. ; FISHER, R.: A Buyer's Guide to Conic Fitting, 1995, S. 513–522
- [FF02] FABER, Pteko ; FISHER, Robert: Estimation of General Curves and Surfaces to Edge and Range Data by Euclidean Fitting / Division of Informatics, University of Edinburgh. 2002. – Forschungsbericht
- [For03] FORSYTH, David: *Computer Vision A Modern Approach*. Pearson Education Inc, 2003
- [Fre05] FRESE, Udo: *Vorlesung Echtzeitbildverarbeitung SS 05*. Universität Bremen. 2005. – Erhältlich unter: <<http://www.informatik.uni-bremen.de/~ufrese/teaching/ebv05/index.d.html>>
- [GGS96] GANDER, W. ; GOLUB, G. H. ; STREBEL, R.: Least-Squares Fitting of Circles and Ellipses. In: EDITORIAL BOARD BULLETIN BELGIAN MATHEMATICAL SOCIETY (Hrsg.): *Numerical analysis (in honour of Jean Meinguet)*, 1996, S. 63–84
- [HF98] HALIR, R. ; FLUSSER, J. *Numerically Stable Direct Least Squares Fitting of Ellipses*. 1998

- [Hou62] HOUGH, P.V.C.: Method and means for recognizing complex patterns / U.S. Patent 3069654. 1962. – Forschungsbericht
- [HS88] HARRIS, Chris ; STEPHENS, Mike: A Combined Corner and Edge Detector. In: *Proceedings of The Fourth Alvey Vision Conference*. Manchester, 1988, S. 147–151
- [HY04] HE, X.C. ; YUNG, N.H.C.: Curvature scale space corner detector with adaptive threshold and dynamic region of support. In: *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on* Bd. 2, 2004. – ISSN 1051–4651, S. 791–794 Vol.2
- [Inv02] INVERSO, Samuel: Ellipse Detection Using Randomized Hough Transform / <http://www.cs.rit.edu/~sai6189/vision/final/>. 2002. – Forschungsbericht
- [Jäh02] JÄHNE, Bernd: *Digitale Bildverarbeitung*. 5. Auflage. Springer, 2002
- [Jia02] JIANG, X.: *Vorlesungsskript Bildanalyse, Vertiefungsvorlesung SS2002, Technische Universität Berlin, Fakultät IV-Elektrotechnik und Informatik*, 2002
- [Jia05] JIANG, Prof. Dr. X.: Comparison of Interest Point Detectors / Westfälische Wilhelms-Universität Münster. 2005. – Forschungsbericht
- [JY04] JINYANG, Du ; YUMEI, Zhang: Automatic extraction of contour lines from scanned topographic map. In: *Geoscience and Remote Sensing Symposium, 2004. IGARSS '04. Proceedings. 2004 IEEE International* Bd. 5, 2004, S. 2886–2888 vol.5
- [KN98] KAWAGUCHI, T. ; NAGATA, R.-I.: Ellipse detection using a genetic algorithm. In: *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on* Bd. 1, 1998, S. 141–145 vol.1
- [Kov00] KOVESI, P. D.: *MATLAB and Octave Functions for Computer Vision and Image Processing*. School of Computer Science & Software Engineering, The University of Western Australia. 2000. – Erhältlich unter: <http://www.csse.uwa.edu.au/~pk/research/matlabfns/>
- [Kov02] KOVESI, Peter: Edges Are Not Just Steps. In: *ACCV2002: The 5th Asian Conference on Computer Vision*, 2002
- [LH90] LEUNG, M.K. ; HUANG, T.S.: Detecting wheels of vehicle in stereo images. In: *Pattern Recognition, 1990. Proceedings., 10th International Conference on* Bd. i, 1990, S. 263–267 vol.1
- [Luh00] LUHMANN, Thomas: *Nahbereichsphotogrammetrie*. Herbert Wichmann Verlag Heidelberg, 2000
- [MR98] McLAUGHLIN ; ROBERT: Randomized Hough Transform: Improved Ellipse Detection with Comparison / University of Western Australia. 1998. – Forschungsbericht

- [Ots79] OTSU, N.: A threshold selection method from gray level histograms. In: *IEEE Trans. Systems, Man and Cybernetics* 9 (1979), März, S. 62–66. – minimize inter class variance
- [PAA⁺87] PIZER, Stephen M. ; AMBURN, E. P. ; AUSTIN, John D. ; CROMARTIE, Robert ; GESELOWITZ, Ari ; GREER, Trey ; ROMENY, Bart Ter H. ; ZIMMERMAN, John B.: Adaptive histogram equalization and its variations. In: *Comput. Vision Graph. Image Process.* 39 (1987), Nr. 3, S. 355–368. – ISSN 0734–189X
- [PFF96] PILU, M. ; FITZGIBBON, A.W. ; FISHER, R.B.: Ellipse-specific direct least-square fitting. In: *Image Processing, 1996. Proceedings., International Conference on* Bd. 3, 1996, S. 599–602 vol.3
- [SB97] SMITH, Stephen M. ; BRADY, J. M.: SUSAN - A New Approach to Low Level Image Processing. In: *Int. J. Comput. Vision* 23 (1997), Nr. 1, S. 45–78
- [Tau91] TAUBIN, Gabriel: Estimation of Planar Curves, Surfaces, and Nonplanar Space Curves Defined by Implicit Equations with Applications to Edge and Range Image Segmentation. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 13 (1991), Nr. 11, S. 1115–1138. – ISSN 0162–8828
- [XOK90] XU, L. ; OJA, E. ; KULTANEN, P.: A new curve detection method: randomized Hough transform (RHT). In: *Pattern Recogn. Lett.* 11 (1990), Nr. 5, S. 331–338. – ISSN 0167–8655
- [Yu95] YU, W.H.W.: Memory-efficient circle and ellipse detection algorithm in digital images. In: *Systems, Man and Cybernetics, 1995. 'Intelligent Systems for the 21st Century'., IEEE International Conference on* Bd. 5, 1995, S. 4262–4267 vol.5
- [ZL03] ZHANG, Si-Cheng ; LIU, Zhi-Qiang: A new algorithm for real-time ellipse detection. In: *Machine Learning and Cybernetics, 2003 International Conference on* Bd. 1, 2003, S. 602–607 Vol.1
- [Zui94] ZUIDERVELD, Karel: Contrast limited adaptive histogram equalization. (1994), S. 474–485. ISBN 0–12–336155–9