

Automatische Spiegelkalibrierung mit Optimierungsverfahren

Quang Hien Mike Bui
(28208716)

Gutachter: Prof. Dr. Kurt Geihs
Betreuer: Andreas Witsch

31. Oktober 2013

Kurzzusammenfassung

Diese Arbeit beschäftigt sich mit einer Methode, mit welcher Distanzen für ein omnidirectionales Kamerasystem bestimmt werden kann. Dazu wurde ein Verfahren zur Erreichung der automatisierten Kalibrierung der Kamera und des Spiegels entwickelt. Dies wurde mit Hilfe eines parametrisierten Kamera-Spiegel Modells erreicht, dessen Parameter durch ein Optimierungsverfahren bestimmt wurden. Die Parameter wurden bezüglich des Fehlers der Berechnung mit Hilfe des Resilient Propagation Verfahren optimiert. Durch die Optimierung der Parameter ist gewährleistet, dass die Distanzen korrekt sind, obwohl der Spiegel verschoben oder geneigt ist. Die Parameter wurden bezüglich des Fehlers der Berechnung mit Hilfe des Resilient Propagation Verfahren optimiert. Die Ergebnisse zeigen, dass der Fehler etwa 2 mm pro Distanz beträgt und die weiten Distanzen durch die Form des hyperbolischen Spiegels verzerrt und schwer zu ermitteln sind.

Inhaltsverzeichnis

1	Einleitung	3
1.1	Projektbeschreibung	3
1.2	Verwandte Arbeiten	4
1.3	Aufbau des Kamerasystems	5
2	Umsetzung	7
2.1	Bestimmung des Spiegel und Kameramodells	7
2.2	Modellierung der Spiegelung	8
2.3	Optimierungsverfahren	12
3	Evaluation	15
4	Zusammenfassung	17
4.1	Ausblick	17
	Abbildungsverzeichnis	19
	Literaturverzeichnis	21

1 Einleitung

Die Universität Kassel nimmt seit einigen Jahren mit dem Fußballroboterteam Carpe-Noctem an dem RoboCup teil. Der RoboCup ist eine Forschungsinitiative, die mit Hilfe von autonom fußballspielenden Robotern die Forschung im Gebiet der Robotik fördert. Ziel des RoboCups ist es im Jahre 2050 den amtierenden menschlichen Fußballweltmeister mit Robotern zu schlagen. Das Regelwerk ähnelt dem FIFA Regelwerk. Ausnahmen bilden unter anderem die Feldgröße und die Anzahl der Feldspieler. Die Maße eines Fußballroboters sind im Regelwerk festgehalten und dürfen eine Größe von 52x52x80 cm nicht überschreiten. Weiterhin darf der Roboter nicht mehr als 40 kg wiegen. Gespielt wird mit einem FIFA Ball und die Spieldauer beträgt 15 Minuten pro Halbzeit. Die Roboter agieren dabei autonom und müssen anhand ihrer Sensorik und Aktorik kooperierend spielen. Als Sensoren kommen unter anderem Kameras, Kompass und Gyroskope zum Einsatz. Diese müssen vor einem Turnier aufwendig kalibriert werden. Des Weiteren werden Absprachen untereinander über eine drahtlose Netzwerkschnittstelle realisiert.

1.1 Projektbeschreibung

Der wichtigste Sensor eines Middle-Size-League Roboters ist die Kamera. Ohne eine richtig kalibrierte Kamera ist es schwieriger sich selbst zu lokalisieren, oder den Ball zu finden. Während den Wartungs- und Reparaturarbeiten an den Robotern können viele Veränderungen an den Robotern geschehen. Auch während eines Transports der Roboter können sich die Positionen der einzelnen Module, Halterungen und Sensoren verändern.

Die Aufgabe für das Projekt war es, eine Methode zu entwickeln, um die Kalibrierung des omnidirektionalen Kamerasystems zu erleichtern. Dies bedeutet ein Verfahren zu finden, um eine automatische Kalibrierung der Kamera und des Spiegels zu erreichen. Das hier entwickelte Verfahren benötigt als Eingabedaten Messpunkte. Die Messpunkte sollten einem Tupel aus Distanz und Pixelwert und einem dessen zugeordneten Winkel entsprechen. Das bedeutet, dass an einem gewissen Pixel in der Bildebene, eine dazugehörige Distanz im Raum mit einem gewissen Winkel im Bild gemessen wird. Die restlichen Werte werden anhand des Kameramodells bestimmt. Die Parameter des Kameramodells werden anhand der gegebenen Werte optimiert.

Vor dieser Projektarbeit wurden die Distanzen an den entsprechenden Pixeln per Hand kalibriert. Dieser Prozess ist sehr aufwändig und meist ungenau. Weiterhin ist es schwierig ohne Domänenwissen über die freien Parameter einen Spiegel korrekt zu kalibrieren. Somit können die Distanzen nur von erfahrenden Mitgliedern des Roboterteams kalibriert werden.

In dieser Arbeit wurden die Werte anhand des Spiegelmodells des hyperbolischen Spiegels bestimmt. Somit sind diese Werte, vorausgesetzt die Vorbedingungen und Parameter sind richtig eingestellt, nahezu optimal. Es kann durch den Einsatz eines Optimierungsverfahren ein lokales Optima entstehen, was bedeuten würde, dass die Parameter nicht das globale Optima abbilden.

Die Herausforderung in diesem Projekt war es, aus der Funktion des Hyperbols ein Modell für den Spiegel inklusive Kameramodell zu erzeugen. Weiterhin soll das Modell mit Hilfe von möglichst wenigen eingemessenen Werten optimiert werden, um die restlichen Pixel zu Distanzwerten zu berechnen. Das Ziel der Optimierung ist es, unabhängig der Position des Spiegels, zuverlässige Distanzwerte zu ermitteln. Somit kann der Spiegel in jede Achse gekippt sein und das Verfahren bestimmt trotz alledem die Distanzen bezüglich eines Pixels in der Pixelebene. Jedoch muss als Trade-off angesehen werden, dass die zu sehende Distanz sich bei einem schief eingestellten Spiegel verringert.

1.2 Verwandte Arbeiten

Um einen omnidirektionalen Spiegel zu kalibrieren gibt es viele Möglichkeiten die zum Einsatz kommen können. Es können 5 verschiedene Klassen von Kalibrierungen erwähnt werden.

1. Selbst-Kalibrierung ohne Szenen Informationen. Kang [3] nutzt die Beschaffenheit von paarweise verketteten Punkt-Features für die Kalibrierung. Diese Methode ist aber nur für omnidirektionale Kameras mit paraboloiden Spiegeln anwendbar.
2. Die „spährenbasierte Kalibrierung“ wurde von Ying und Hu [6] eingeführt. Sie analysierten die Relation zwischen den intrinsischen Kameraparametern und der im Bereich abgebildeten Kontur. Anhand dieser Relation war es ihnen möglich eine omnidirektionale Kamera zu kalibrieren.
3. Linienbasierte Kalibrierung. Geyer und Daniilidis [1] haben in ihrer Arbeit gezeigt, dass sie mit Hilfe von mindestens drei Linienbildern eine omnidirektionale Kamera kalibrieren können. Sie untersuchten die geometrischen Eigenschaften von Linienbildern unter dem Kamera-Spiegel-Modell einer omnidirektionalen Kamera.
4. Punktbasierte Kalibrierung. Bei dieser Methode werden bekannte 3-dimensionale Punkte im Raum genutzt, um die Parameter des Spiegel-Kameramodells zu bestimmen.

5. 2D Kalibrierungsmethode. Kannate und Brandt [4] zeigten in Ihrer Arbeit eine Methode Fish-Eye und omnidirektionale Kameras anhand von 2-dimensionalen Mustern zu kalibrieren. Dadurch, dass ein 2-dimensionales Muster das ganze Kamera Bild abdeckt, werden mit dieser Methode sehr gute Kalibrierungen erzielt.

Diese Arbeit beschäftigt sich mit der Punktbasierter-Kalibrierungsmethode. Auch hier werden Punkte im 3-dimensionalen Raum gemessen. Jedoch beschreibt die Messung in dieser Arbeit nur eine Distanz zu Pixel in der Bildebene Mapping. Anhand dieser Messungen werden die Parameter des Kamera und Spiegelmodells optimiert.

1.3 Aufbau des Kamerasystems

Der aktuelle Aufbau des Kamerasystems besteht aus einer Kamera, die vertikal auf einen hyperbolischen Spiegel zeigt. Der Spiegel reflektiert ein rundum Bild von der Umgebung, welches von der Kamera aufgenommen wird. Der Spiegel und die Kamera sind am oberen Teil des Roboters angebracht und befinden sich in etwa 60 cm über dem Boden. Der Spiegel kann die Umgebung in einem weiten Radius aufnehmen. Durch die hyperbolische Form des Spiegels sind die Objekte in weiter Entfernung, also am Rand des Spiegels sehr verzerrt.

Eine weitere Eigenschaft, welche durch die Form des Spiegels entsteht ist, dass die Distanz vom Mittelpunkt bis zum Rand exponentiell ansteigt. Das bedeutet, dass der Distanzunterschied von einem Pixel zum anderen größer wird, je weiter sich der Pixel dem Rand des Spiegels annähert.

2 Umsetzung

In diesem Kapitel werden die Ansätze und die Umsetzung der Kamera-Spiegel-Kalibrierung diskutiert. Zu Beginn wird der Ansatz zur Bestimmung des Spiegel- und Kameramodells erläutert. Danach folgt die Umsetzung und die Optimierung des Modells.

2.1 Bestimmung des Spiegel und Kameramodells

Zu Beginn der Arbeit wurde einzig die 3-dimensionale Funktion des Spiegels gegeben. Die Funktion 2.1 beschreibt einen 3-dimensionalen Hyperbol. Um eine Distanztabelle, unabhängig von der Position des Spiegels, bestimmen zu können, müssen die Positionsparameter in diese Funktion hinein modelliert werden. Dies bedeutet, dass für die X, Y und Z Neigung ein Parameter bestimmt werden muss, um die Funktionen entsprechend der Neigung zu bestimmen. Zur Vereinfachung der Berechnung der Parameter wurde die 3-dimensionale Funktion in eine 2-dimensionale Funktion überführt.

$$\frac{z^2}{789.3274} - \frac{x^2 + y^2}{548.144} = 1 \quad (2.1)$$

Die Überführung wurde erreicht, in dem $x^2 + y^2$ durch $r = x^2 + y^2$ substituiert wurde. Somit wurde die Funktion auf die z-r-Ebene projiziert. Diese Hyperbel stellt einen Schnitt dar, welcher bei Rotation um die Z-Achse die Hyperboloidfläche erzeugt. Dies bedeutet für die Umsetzung der Aufgabenstellung, dass jeweils nur eine Gerade in der Pixelebene für einen gegebenen Winkel betrachtet wird.

$$f(x) = \sqrt{789.3274 \cdot \left(1 + \frac{x^2}{548.144}\right)} \quad (2.2)$$

Der Winkel beschreibt die Rotation einer Geraden in der Bildebene. Somit kann durch die Rotation der Geraden jeder Pixel der Bildebene einzeln durch die 2-dimensionale Spiegelfunktion bestimmt werden. Um das vollständige Modell aus Spiegel und Kamera zu bestimmen, muss am Anfang bestimmt werden welcher Pixel in der Bildebene welchen Schnittpunkt an der Hyperbel besitzt und welcher Schnittpunkt mit dem Boden daraus resultiert. Um dies zu bestimmen muss für jeden Pixel ein Strahl modelliert werden, der am Spiegel gespiegelt wird.

2.2 Modellierung der Spiegelung

Die Modellierung des Spiegels wird nicht wie vermutet aus der Spiegelung der Raumkoordinate bestimmt, sondern ausgehend aus der Bildebene. Das bedeutet, man konstruiert aus der Bildebene einen Strahl und bestimmt dann einen Schnittpunkt mit dem Boden. Durch die Annahme, dass die Kamera einem Lochkamera Modell unterliegt, schneidet jeder Strahl, der auf die Bildebene trifft einen definierten Fokuspunkt. Somit wird im ersten Schritt eine Gerade mit einem bestimmten Pixel als Stützvektor durch den Fokuspunkt konstruiert.

Dadurch, dass die Gerade in der Pixelebene in Pixeln und außerhalb der Pixelebene in mm bzw. cm bestimmt werden muss, muss hier für die Bestimmung der Geraden zuerst mit Hilfe des Strahlensatzes der Winkel α wie in Abbildung 2.1 zu sehen ist bestimmt werden. Das bedeutet es wird der Rand des Spiegels als äußerste Kante gewählt. Dieser Punkt wird auch auf der Bildebene zu erkennen sein.

Im Detail kann die Bestimmung des Kameramodells in mehrere Schritte unterteilt werden:

1. Bestimmung der Anfangsparameter
2. Ausgangsstrahl von der Bildebene bestimmen
3. Ausgangsstrahl, vom Fokuspunkt aus, mit dem gleichen Winkel wie in 2.) konstruieren
4. Ausgangsstrahl am Spiegelschnittpunkt spiegeln
5. Neuen gespiegelten Ausgangsstrahl mit dem Boden schneiden und die Distanz vom Mittelpunkt bis zum Schnittpunkt bestimmen

Einer der unbekanntenen Anfangsparameter, der bestimmt werden muss, ist die Höhe vom Fokuspunkt in der Bildebene. Um dies zu bestimmen, wird wie in Abb 2.1 gezeigt ein Dreieck zwischen dem Rand des Spiegels und dem Spiegelmittelpunkt, sowie dem Fokuspunkt konstruiert. Die Winkel, die dort im Dreieck bestimmt werden, lassen sich dann auch in der Bildebene wiederfinden. Es ist möglich die Längen der jeweiligen Seite mit Hilfe des Strahlensatzes zu bestimmen, doch kann mit diesen Werten nicht weiter gerechnet werden, da die Werte innerhalb der Bildebene in Pixeln angegeben sind und außerhalb der Pixelebene in mm bzw. cm. Anhand der Winkel und der Information, wie groß der Radius der Bildebene ist, lässt sich die Höhe h' bestimmen. Diese wird benötigt um den Ausgangsstrahl von der Bildebene zu konstruieren, welches im nächsten Schritt folgt.

Nun wird der Ausgangsstrahl von der Bildebene bestimmt. Um das Mapping von einem Pixel in der Bildebene zu einer Distanz im Raum zu bekommen, muss von der Bildebene aus, der Ausgangsstrahl bestimmt werden. Das bedeutet, es wird eine Gerade konstruiert, die von dem zu berechnenden Pixelwert ausgeht und durch den Fokuspunkt geht. Diese Gerade sollte nun den Spiegel schneiden. Um die Berechnung unabhängig der Einheiten zu gestalten, wurde der Winkel α vom Ausgangsstrahl bestimmt und ein Punkt um den Fokuspunkt um den

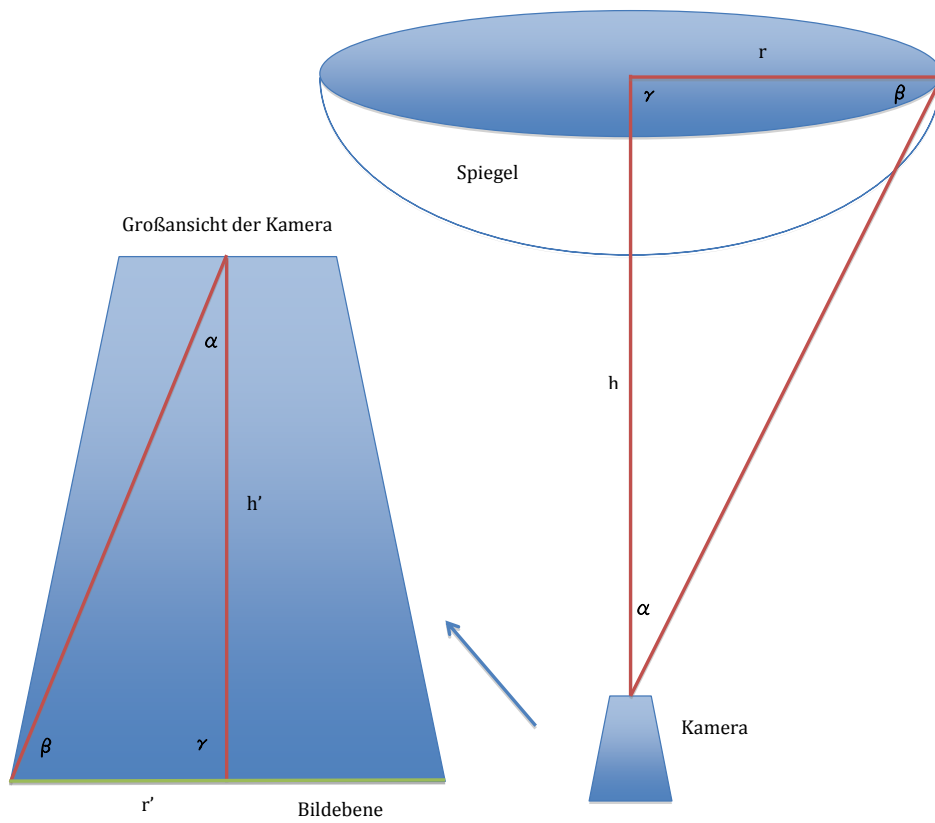


Abbildung 2.1: Bestimmung des Ausgangsstrahl

berechneten Winkel rotiert. Danach wird eine Gerade konstruiert die durch den Fokuspunkt und durch den neu rotierten Punkt geht. Um den Winkel zu bestimmen wurde folgende Formel genutzt:

$$\alpha = \frac{180}{\pi} \cdot \arctan\left(\frac{x}{h}\right) \quad (2.3)$$

Mit x als die Anzahl von Pixel vom Mittelpunkt der Bildebene bis zum berechneten Pixel und h als die Höhe von der Bildebene bis zum Fokuspunkt.

Abbildung 2.2 zeigt den Spiegel und die konstruierte Gerade ohne die Kamera. Die Gerade wird vom Fokuspunkt aus konstruiert und die X-Achse bildet in der Abbildung den Boden des realen Systems ab.

Im nächsten Schritt wird der Schnittpunkt der Gerade mit dem Spiegel bestimmt. Dadurch das im Vorverarbeitungsschritt die 3-dimensionale Spiegelfunktion auf eine 2-dimensionale Spiegelfunktion projiziert wurde, wird der Schnitt durch zwei Werte repräsentiert. Um den Schnittpunkt zu bestimmen muss die Ausgangsgerade mit der Spiegelfunktion gleichgesetzt und aufgelöst werden.

$$\sqrt{789.3274 \cdot \left(1 + \frac{x^2}{548.144}\right)} + o = b + \frac{x - a}{c} \cdot d \quad (2.4)$$

Die Variablen a, b, c, d beschreiben die Stütz- und Richtungsvektoren der Ausgangsgeraden:

$$\begin{pmatrix} a \\ b \end{pmatrix} + \lambda \cdot \begin{pmatrix} b \\ c \end{pmatrix} \quad (2.5)$$

Um nun die Gerade mit Hilfe der Spiegelfunktion zu spiegeln, wird die Tangente am Schnittpunkt gebildet. Dazu muss die Steigung der Tangente bestimmt werden, die durch die erste Ableitung der Spiegelfunktion am Schnittpunkt bestimmt werden kann. Weiterhin wird eine Senkrechte, die durch den Schnittpunkt geht, gebildet und ein Punkt wird auf der zu spiegelnden Gerade an der Senkrechten gespiegelt. Die gespiegelte Gerade wird dann durch den Schnittpunkt und durch den neu gespiegelten Punkt berechnet. Diese Gerade wird dann im letzten Schritt mit dem Boden geschnitten und die Distanz vom Schnittpunkt zum Mittelpunkt des Roboters bestimmt (Abbildung 2.2).

Dieses Verfahren wird für jeden Pixel in der Bildebene wiederholt, wobei immer der 2-dimensionale Fall betrachtet wird und die Gerade auf der Bildebene bei jedem Schritt rotiert wird. Abbildung 2.3 zeigt einige Beispielgeraden, die am Spiegel gespiegelt wurden.

Um die Neigung des Spiegels zu modellieren wurden drei Parameter in Prozess der Berechnung mit hinein modelliert. Es wurde beim Spiegeln der Eingangsgerade an der Hyperbel eine Offset-Variable modelliert, welche die Neigung und die Höhenänderung des Spiegels

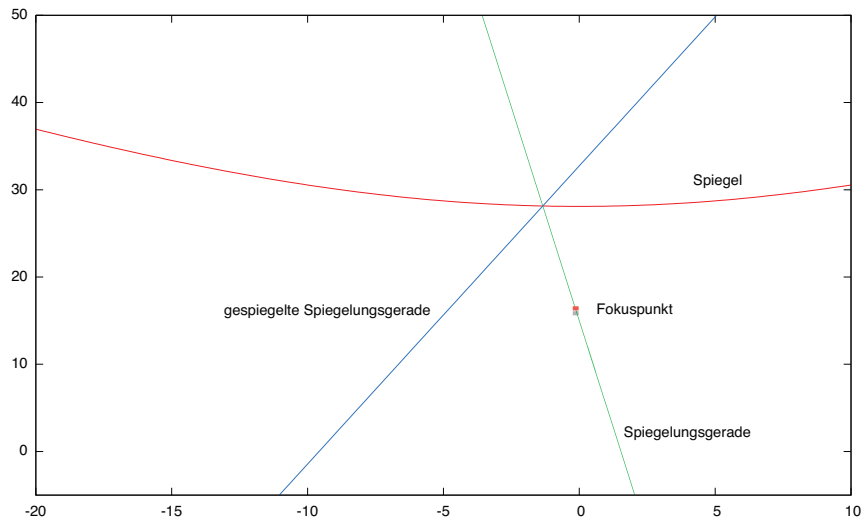


Abbildung 2.2: Ausgangsstrahl als gespiegelte Spiegelungsgerade am Spiegel

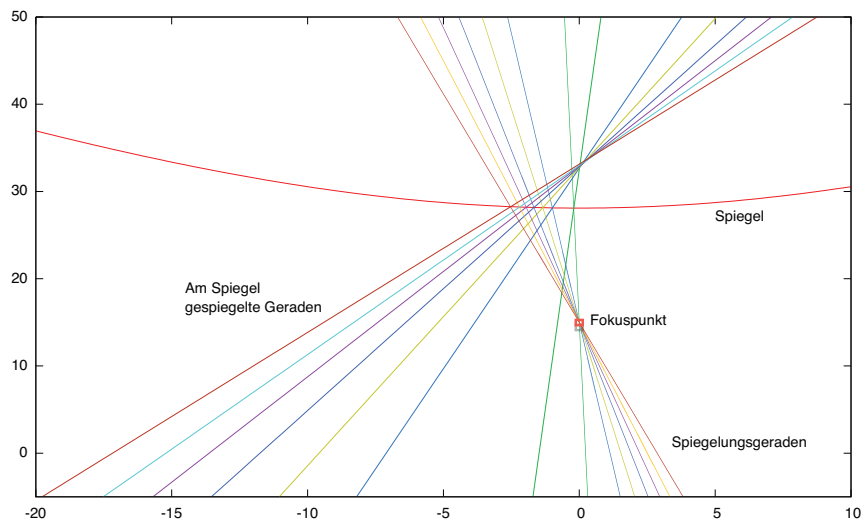


Abbildung 2.3: Beispielspiegelungen

modelliert. Weiterhin wurde das Ergebnis der Distanzfunktion mit einer Geradenfunktion modelliert. Das bedeutet, dass die Werte am Ende der Berechnung noch durch eine Geradenfunktion der Form $f(x) = m \cdot x + b$ modelliert werden. Der letzte Parameter ist eine Offset-Variable für den Winkel, welche die Verschiebung des Spiegels modellieren soll.

2.3 Optimierungsverfahren

Die neu modellierten Parameter müssen im nächsten Schritt optimiert werden. Dazu werden die gemessenen Datensätze benötigt. Ziel eines Optimierungsverfahren ist es, einen bestimmten Parametersatz soweit zu optimieren, dass die Zielfunktion minimiert oder maximiert wird. In diesem Fall sollen die Parameter soweit optimiert werden, dass die jeweiligen Spiegelungsgeraden von einem Pixel, die dazugehörige gemessene Distanz so gut wie möglich annähern. Dies bedeutet, dass der Fehler von einem zu dem nächsten Schritt minimiert werden muss.

Es können zur Optimierung der Parametersätze viele Optimierungsverfahren eingesetzt werden (vgl. CMA-ES [2], PSO [5]). Das meist bekannteste Optimierungsverfahren ist das Gradientenabstiegsverfahren. Weitere Optimierungsverfahren sind beispielsweise das Gauß-Newton Verfahren, das Levenberg-Marquadt Verfahren und das Resilient Propagation Verfahren welche auf dem Gradientenabstiegsverfahren basieren. Alle Optimierungsverfahren haben gemeinsam, dass sie mit einem initialen Parametersatz starten. Für diesen Parametersatz wird nun der Fehler berechnet und es wird in Richtung des negativen Gradienten abgestiegen um ein lokales Optimum zu erreichen. Dies wird so lange wiederholt bis das Verfahren konvergiert.

Zur Berechnung des optimalen Parametersatzes wurde in dieser Arbeit das Resilient Propagation (RProp) Verfahren eingesetzt.

$$\gamma_{ij}^{(k+1)} = \begin{cases} \min(\gamma_{ij}^{(k)} \cdot \eta^+, \gamma_{max}) & \text{falls } \nabla_{ij} E^{(k)} \cdot \nabla_{ij} E^{(k-1)} > 0 \\ \max(\gamma_{ij}^{(k)} \cdot \eta^-, \gamma_{min}) & \text{falls } \nabla_{ij} E^{(k)} \cdot \nabla_{ij} E^{(k-1)} < 0 \\ \gamma_{ij}^{(k)} & \text{sonst} \end{cases} \quad (2.6)$$

Bewährte Parametersätze sind $\eta^+ = 1,2$, $\eta^- = 0,5$, $\gamma_{max} = 50$, $\gamma_{min} = 10^{-6}$. Mit η als die Schrittweite der Veränderung und γ als den gewichtsindividuellen Parameter für die Schrittweite.

Mit dem RProp Verfahren wird nun iterativ für jeden gemessenen Winkel ein optimaler Parametersatz bestimmt. Dabei wird für jeden Winkel so lange ein Parametersatz verändert, bis der Fehler konvergiert. Dabei wird mit einem initialen Parametersatz angefangen. Dieser Parametersatz entspricht einem optimal eingestellten Spiegel. Nun wird dieser Parametersatz

anhand des Gradienten verändert. Mit diesem neu errechneten Parametersatz wird erneut die Funktion berechnet und der Fehler ermittelt. Dies wird so lange wiederholt, bis der Fehler konvergiert. Das bedeutet, dass der Gradient durch die Differenz der berechneten Fehler bestimmt wird. In jedem Iterationsschritt wird eine Parametervariation erzeugt. Mit diesen neu variierten Parametern wird die Distanz bestimmt und danach der Chi-Square-Error des aktuellen Distanzwerts und des vorhergegangenen Distanzwert bestimmt. Durch die Differenz des Fehlers kann der Gradient bestimmt werden, mit dessen Hilfe die Schrittänderung des RProp Verfahrens bestimmt wird. Der Chi-Square-Error wird folgendermaßen berechnet:

$$CSE = \frac{\sum_{i=1}^n (a_i - b_i)^2}{2} \quad (2.7)$$

Falls die Differenz der Fehler einen gewissen Grenzwert nicht mehr überschreiten, ist dies ein Indiz für die Konvergenz des Optimierungsverfahrens und es wird der aktuelle Parametersatz als optimierter Parametersatz definiert. Anhand des Parametersatzes können nun die entsprechenden Distanzen zum dazugehörigen Winkel bestimmt werden. Für die anderen nicht gemessenen Winkel, werden die am nächsten dazu gehörigen Winkel betrachtet, welches die benachbarten gemessenen Winkel darstellen. Um die exponentielle Steigung besser bestimmen zu können, werden nur Werte innerhalb von einem Meter interpoliert. Das bedeutet, dass für einen Winkel X die gemessenen Nachbarwinkel betrachtet werden. Dann wird für das Intervall 0 bis 1 Meter ein Parametersatz bestimmt und die dazugehörigen Distanzen bestimmt. Danach wird für das Intervall 1 bis 2 Meter ein neuer Parametersatz bestimmt und die Distanzen berechnet. Dies wird solange fortgeführt, bis alle Distanzen abgedeckt sind.

Dieses Verfahren benötigt eine längere Laufzeit, doch sind die zu bestimmenden Distanzen besser als wenn ein Parametersatz für die ganze Gerade in der Pixelebene bestimmt wird. Dies lässt sich anhand der Funktion erkennen, da die Funktion exponentiell steigt. Somit würde ein Parametersatz über die ganze Funktion dies schlecht abdecken. Doch viele Parametersätze für die Intervalle interpolieren die Funktionswerte recht gut.

3 Evaluation

Auf der WM 2013 in Eindhoven wurde eine Distanzdatei eingemessen. Eine Distanzdatei besteht aus mehreren Winkeln, zu denen jeweils in einem Meter Abstand zu einander, Punkte gemessen worden sind. Zu den jeweiligen Distanzen sind die dazu gehörigen Pixel eingetragen. Zum Einmessen wurde ein schmaler Teppich ausgerollt, auf dem im Abstand von 1 Meter Markierungen aufgeklebt waren. Der Roboter wurde auf diesen Teppich platziert und um die eigene Achse gedreht. Es wurden bei jedem Winkel die Markierungen detektiert und die Distanz zum Mittelpunkt des Robotermittelpunkts bestimmt. Der Pixelwert, an dem die Markierung gemessen wurde, wurde eingetragen. Somit entstand für jeden Winkel im Bild eine Gerade, auf der die 1 Meter Markierungen zu erkennen sind.

Diese Datei wurde für die Evaluation modifiziert, indem einige Winkel oder speziell einige Werte herausgelöscht wurden. Um eine anschaulichere Evaluation zu bekommen, wurden nur Distanztabelle der Größe 50x50 Pixel erzeugt. Dies bedeutet, dass die Bildebene auf 50x50 Pixel reduziert wurde. Es wurden zufällig Winkel und Distanzwerte entfernt. Im besten Fall sollten, obwohl einige Winkel und Distanzwerte fehlen, für die jeweiligen Pixel die gleichen Distanzen errechnet werden können. Um dies zu überprüfen, wurde jeweils für die originale und die modifizierten Distanzdateien eine 50x50 große Distanztabelle erzeugt. Danach wurden die Fehler bezüglich der Distanzen bestimmt. Anhand des Fehlers kann bestimmt werden, wie gut das Optimierungsverfahren gearbeitet hat. Es wurden für die Evaluation 5 Modifikationen an der Distanzdatei durchgeführt.

Beim Vergleich der Fehler in Tabelle 3.1 der verschiedenen modifizierten Distanzen ist zu sehen, dass der minimale Fehler bei etwa 1,8 Meter und der maximale Fehler bei 9 Metern liegt. Dieser Fehler beschreibt den gesamten Fehler des Verfahrens. Somit entstand bei 2500 Distanzwerten eine Abweichung zwischen 1,8 und 94 Metern. Das wäre im Minimalfall eine Abweichung von 0,22 mm pro Pixel und im Maximalfall eine Abweichung von 3,7 mm pro Pixel. Das bedeutet, dass der Fehler sehr gering ist und somit das Optimierungsverfahren sehr stabil und zuverlässig arbeitet. Somit ist die Schätzgenauigkeit des Verfahrens bei unter 5 mm pro Pixel. Das heißt, dass der wahre Distanzwert um unter 5 mm pro Pixel abweicht und somit für große Distanzen vernachlässigbar ist. Im Durchschnitt liegt der Fehler bei etwa 5,16 m mit einer Varianz von 6,88 m und einer Standardabweichung von 2,62 m. Dies entspricht einer Abweichung von 2 mm pro Pixel.

Über die Rechenzeit der Optimierung ist zu sagen, dass die Optimierung einer vollständigen

Modifikationen	Gesamtfehler
Modifikation 1	9.43 Meter
Modifikation 2	4.65 Meter
Modifikation 3	6.51 Meter
Modifikation 4	1.86 Meter
Modifikation 5	3.36 Meter

Tabelle 3.1: Tabelle mit Gesamtfehlern der Evaluation

Distanztabelle für ein Roboter auf einem Windows 7 Laptop mit Core i5 Prozessor etwa 20 Minuten benötigt. Diese Zeit ist abhängig von der Größe der zu berechnenden Distanzmatrix. Weiterhin kann die Berechnungszeit verringert werden, wenn eine andere Methode der Parameterbestimmung gewählt wird. Durch die Aufteilung der jeweils 1 Meter Intervalle müssen mehr Iterationen für die Berechnung der Parameter getätigt werden. Durch die Berechnung einer Zeile der Matrix mit nur einem Parametersatz kann die Rechenzeit reduziert werden. Doch ist diese Berechnung nicht genau und kann somit als Trade-Off angesehen werden.

4 Zusammenfassung

Durch die automatische Spiegelkalibrierung kann die Arbeit während eines Turniers oder nach einer Hardware Reparatur am Kamerabaum deutlich erleichtert werden. Durch den Einsatz eines Optimierungsverfahrens zur Berechnung der Distanzen ist es nicht mehr nötig den Spiegel korrekt einzustellen. Es kann nun auch ein nicht optimal eingestellter Spiegel genutzt werden, da die Fehler mit Hilfe des Optimierungsverfahrens herausgerechnet werden können. Die modellierten Parameter des Kamera-Spiegel Modells erlauben eine Distanzmatrix ohne korrekt eingestellten Spiegel. Mit Hilfe des RProp Optimierungsverfahrens werden die Parameter für die Verschiebung und Neigung des Spiegels, innerhalb des Kamera-Spiegel Modells und abhängig der Neigung und Verschiebung des Spiegels bestimmt. Es müssen nun nicht mehr der Spiegel und die Kamera von Hand kalibriert werden um eine gute Distanztabelle zu erzeugen.

Die Evaluation hat gezeigt, dass der Fehler des Optimierungsverfahren bei schlechten oder fehlerhaften Messdaten vernachlässigt werden kann. Die Abweichung kann vernachlässigt werden, da die Abweichung etwa 2 mm pro Pixel beträgt.

Leider konnte aus zeitlichen Gründen dieser Ansatz während der Weltmeisterschaft 2013 in Eindhoven nicht eingesetzt werden. Dieser soll jedoch auf dem nächsten Turnier zum Einsatz kommen um die Verbesserung zum vorherigen Ansatz beweisen zu können.

4.1 Ausblick

Abschließend ist zu erwähnen, dass noch viel Arbeit in die Thematik der automatischen Spiegelkalibrierung investiert werden kann. Beispielsweise wäre es hilfreich für den ganzen Kalibrierprozess, wenn die Distanzdatei automatisiert erstellt werden könnte. Somit könnte der Kalibrierprozess völlig automatisiert ablaufen und es würde eine große Zeitersparnis für ein Turnier sein. Weiterhin kann das Optimierungsverfahren in späteren Arbeiten verbessert werden. Zum derzeitigen Stand benötigt die Optimierung einer Distanzdatei bis zu etwa 20 Minuten. Mit Hilfe von effizienteren Optimierungsverfahren könnte wahrscheinlich die Berechnungszeit deutlich minimiert werden. Doch die größte Verbesserung wird mit der Verbesserung der Intervalle und einer verbesserten Organisation der Datenstrukturen erreicht. Im aktuellen Quellcode wird mit verketteten Listen gearbeitet. Eine Optimierung der im Quellcode genutzten verketteten Listen hin zu Pointern würde die Rechenzeit verringern.

Abbildungsverzeichnis

2.1	Bestimmung des Ausgangsstrahl	9
2.2	Ausgangsstrahl als gespiegelte Spiegelungsgerade am Spiegel	11
2.3	Beispielspiegelungen	11

Literaturverzeichnis

- [1] GEYER, C. ; DANILIDIS, K.:
Paracatadioptric camera calibration.
In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 24 (2002), Nr. 5, S. 687–695. –
ISSN 0162–8828
- [2] HANSEN, Nikolaus ; MÜLLER, Sibylle D. ; KOUMOUTSAKOS, Petros:
Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES).
In: *Evol. Comput.* 11 (2003), März, Nr. 1, 1–18.
<http://dx.doi.org/10.1162/106365603321828970>. –
DOI 10.1162/106365603321828970. –
ISSN 1063–6560
- [3] KANG, Sing B.:
Catadioptric Self-Calibration.
2000
- [4] KANNATA J, Brandt S.:
A generic camera model and calibration method for conventenal, wide-eye, and fish-eye lenses.
In: *IEEE Transactions on Pattern Analysis and Machine Intel- ligen*ce, 2006, S. 1335–1340
- [5] KENNEDY, James:
Particle swarm optimization.
In: *Encyclopedia of Machine Learning*.
Springer, 2010, S. 760–766
- [6] YING, X. ; HU, Z.:
Catadioptric Camera Calibration Using Geometric Invariants.
In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26 (2004), Oktober, Nr. 10