



INSTITUT FÜR INFORMATIK VII
ROBOTIK UND TELEMATIK

Masterarbeit

**Entwicklung einer randomisierten
Hough-Transformation zur
Zylinderdetektion in Laserscans**

Sarah Menninger

Januar 2021

Erstgutachter: Prof. Dr. Andreas Nüchter
Zweitgutachter: Prof. Dr. Klaus Schilling
Betreuer: Michael Bleier

Zusammenfassung

Neben Ebenen sind Zylinder eine der am häufigsten auftretenden grundlegenden Formen im industriellen Bereich, da sie als geometrische Primitive für Rohrleitungen, Gerüste und dergleichen dienen. Eine schnelle, effiziente und robuste Methode zur Detektion von Zylindern in Punktwolken ist somit essentiell, um die manuellen Schritte bei der Modellerzeugung zu minimieren. Auch wenn es möglich ist, mithilfe einer 5D Hough-Transformation Zylinder zu detektieren, so ist dieser Ansatz doch sehr komplex und zeitintensiv. Aus diesem Grund ist diese Methode auf großen Punktwolken nicht direkt anwendbar. Eine Verringerung der Komplexität kann durch einen sequentiellen zwei Schritt Ansatz ermöglicht werden. Der erste Schritt findet mithilfe einer 2D-Hough-Transformation eine starke Hypothese für die Zylinderachsenorientierung, während sich der Zweite mit der Suche des Radius und der Zylinderposition im dreidimensionalen Raum beschäftigt. Die vorliegende Arbeit stellt eine Randomized Hough-Transformation (RHT) basierte Implementierung vor, die sich damit beschäftigt, möglichst effizient Zylinder und Teilzylinder bis hin zu Halbzylinder automatisch zu detektieren. Mithilfe Simulations- sowie Echtdaten werden die Resultate anschließend verifiziert und bewertet.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Zielsetzung und wissenschaftlicher Beitrag	2
1.3	Aufbau der Arbeit	2
2	Stand der Technik	5
2.1	Zylinderdetektion mit segmentierten Punktwolken	5
2.2	Zylinderdetektion auf unsortierten Punktwolken	6
3	Zylinderdetektion mithilfe der Hough-Transformation	7
3.1	Hough-Transformation	7
3.2	2D Hough-Transformation	8
3.3	Zylinderdetektion	9
3.3.1	Zylinder Parametrisierung	10
3.3.2	Zylinderachsendetektion durch Hough-Transformation	11
3.3.3	3D Hough-Transformation	12
3.3.4	Positions- und Radiusberechnung	13
3.3.5	Randomisierte Hough-Transformation	14
4	Implementierung	17
4.1	Zylinderachsendetektion	18
4.1.1	Vorbereitung - Berechnung sphärischer Normalen	18
4.1.2	Akkumulator	22
4.1.3	RHT Zylinderachsendetektion	24
4.1.4	Generierung eines orthonormalen Koordinatensystems	29
4.2	Positions- und Radiusberechnung	30
4.2.1	Vorbereitung - Selektion wahrscheinlicher Zylinderpunkte	31
4.2.2	Vorbereitung - Projektion der Zylinderpunkte	31
4.2.3	Standard Hough-Transformation	33
4.2.4	Randomized Hough-Transformation	36
4.2.5	Least-Square Kreis Fitting	39
4.2.6	Pseudocode der RHT-Kreisdetektion	42
4.3	Zylinderenddetektion und Validierung detektierter Zylinder	43

4.3.1	Zylinderenddetektion	43
4.3.2	Erkennung von doppelten Zylindern	44
5	Experimente und Auswertung	47
5.1	Auswertung der Kreisdetektion	48
5.1.1	Auswertung Kreisdetektion bei fehlerfreier Zylinderachse und rauschfreier Punktvolke	49
5.1.2	Auswertung der Kreisdetektion bei fehlerbehafteter Zylinderachse und rauschfreier Punktvolke	54
5.2	Experiment und Auswertung der Zylinderdetektion	62
5.2.1	Generierung der Testdaten und Aufteilung der detektierten Zylinder . . .	63
5.2.2	Auswertung Zylinderachsendetektion auf rauschfreier Punktvolke	65
5.2.3	Auswertung Kreisdetektion auf rauschfreier Punktvolke	68
5.2.4	Bewertung Zylinderdetektion auf rauschfreier Punktvolke	71
5.2.5	Generierung realer Scan und Kategorisierung der detektierten Zylinder . .	72
5.2.6	Auswertung Zylinderachsendetektion auf realen Daten	75
5.2.7	Auswertung Kreisdetektion auf realen Daten	79
5.2.8	Bewertung Zylinderdetektion auf realen Daten	82
5.3	Experiment und Auswertung der Detektion von halben Zylindern	82
5.3.1	Generierung der Testdaten und Aufteilung der detektierten Zylinder . . .	83
5.3.2	Auswertung Zylinderachsendetektion auf rauschfreier Punktvolke	86
5.3.3	Auswertung Kreisdetektion auf rauschfreier Punktvolke	90
5.3.4	Bewertung Zylinderdetektion für halbe Zylinder	93
5.3.5	Generierung realer Scan und Kategorisierung der detektierten Zylinder . .	93
5.3.6	Auswertung Zylinderachsendetektion auf realen Daten	97
5.3.7	Auswertung Kreisdetektion auf realen Daten	101
5.3.8	Bewertung Teilzylinderdetektion auf realen Daten	106
5.4	Zusammengefasste Auswertung der Zylinderdetektion	107
6	Zusammenfassung und Ausblick	109

Kapitel 1

Einleitung

Diese Arbeit befasst sich mit der Entwicklung einer randomisierten Hough-Transformation zur Zylinderdetektion in Laserscans. Hierfür wurde ein zweiteiliger sequentieller Algorithmus gewählt, welcher zuerst die Zylinderachsen bestimmt und anschließend diese nutzt, um mithilfe einer Kreisdetektion, Radius und Position des Zylinders im dreidimensionalen Raum zu bestimmen.

1.1 Motivation

3D Laserscanning ist mittlerweile eine aufgrund ihrer Effektivität und Schnelligkeit verbreitete Möglichkeit, um dichte dreidimensionale räumliche Daten, genannt Punktwolken, zu erstellen. Da Zylinder neben Ebenen eine der häufigsten grundlegenden Formen im industriellen Bereich sind, ist die Zylinderdetektion ein grundlegender Bestandteil um ein genaues Modell einer industriellen Szene zu erzeugen [22]. Zylinder können als graphische Primitive für Rohrsysteme oder Tanks verwendet werden, jedoch treten sie somit in verschiedenen Größen und Ausrichtungen auf. Gleichzeitig stellen viele Zylinderdetektionsalgorithmen auf unsortierten Punktwolken häufig mehrere Anforderungen an die Zylinderparameter, etwa was die Achsenausrichtung oder die Radiusgröße betrifft [19]. Aus diesem Grund soll in dieser Arbeit ein Algorithmus vorgestellt werden, welcher Zylinder mit beliebigen Achsen und Radii detektieren kann. Dabei ist es mit dem Ansatz möglich nicht nur ganze, sondern auch Teilzylinder bis hin zu Halbzyindern robust und schnell zu erkennen. Zugleich ist der Algorithmus so gebaut, dass er möglichst einfach erweiterbar für zusätzliche Bedingungen ist, welche helfen können die Laufzeit und Falschdetektionsrate gerade bei Teilzylinderdetektionen zu minimieren.

1.2 Zielsetzung und wissenschaftlicher Beitrag

Das Ziel der Arbeit ist es einen möglichst allgemeinen Zylinderdetektionsalgorithmus auf Grundlage einer randomisierten Hough-Transformation zu entwickeln, wobei keine Einschränkungen des Radiusbereiches oder der Zylinderachsenrichtung notwendig sein sollen. Gleichzeitig soll es auch möglich sein Teilzylinder, bis hin zu Halbzylindern, erfolgreich zu detektieren und schließlich den Algorithmus auf Simulations- sowie Realdaten zu validieren und mögliche Schwächen klar darzustellen.

Ein solcher allgemeingültiger Algorithmus, der keine grundlegenden Anforderungen an die Zylinderparameter stellt, ist ein sinnvoller wissenschaftlicher Beitrag, da sich bisher die meisten Werke lediglich mit Speziallösungen befassen. Diese sind gerade für große Punktwolken schneller, legen jedoch bereits im Vorfeld meist harte Bedingungen für die Zylinderparameter fest, wie etwa vorgegebene Achsenrichtungen ([15]) oder einen festen Radiusbereich ([22], [19]).

Demgegenüber ermöglicht der in dieser Arbeit entwickelte randomisierte Hough-Transformations Ansatz eine Detektion von Zylindern und Teilzylindern, die keine Beschränkung der Parameter erfordert. Die Möglichkeit die Zylinderparameter einzuschränken ist dabei weiterhin gegeben. Somit kann dieser Ansatz unkompliziert erweitert werden, um Falschdetektionen zu Verhindern und seine Laufzeit zu optimieren.

Die Grundidee der Zylinderdetektion mithilfe einer zweiseitigen Hough-Transformation stammt dabei aus dem Artikel [22]. Die eigentliche Implementierung unterscheidet sich jedoch wesentlich, da [22] an verschiedenen Stellen Schwächen aufweist. So ist die Allgemeingültigkeit des Radius nicht gegeben und die Laufzeit kann an vielen Stellen verbessert werden.

Darüber hinaus bietet der in dieser Arbeit vorgestellte Algorithmus weitere Entwicklungen:

- Die Auswertung des Hough-Raumes der Zylinderdetektion erfolgt mithilfe eines Fensterfilters (statt einfachem Threshold).
- Der 2. Schritt der Zylinderdetektion (Kreisdetektion) ermöglicht es mithilfe einer parametrisierbaren Validierungsfunktion gezielt Teilzylinder zu detektieren.
- Das Ergebnis der Kreisdetektion wird mithilfe eines Least-Square Ansatzes präzisiert.
- Die Zylinderenden werden mithilfe der detektierten Mantelflächenpunkte berechnet.

1.3 Aufbau der Arbeit

Die folgende Arbeit teilt sich in die großen Kapitel „Stand der Technik“(2), „Zylinderdetektion mithilfe der Hough-Transformation“(3), „Implementierung“(4), „Experimente und Auswertung“(5), sowie zuletzt der „Zusammenfassung und dem Ausblick“(6). Wobei das Kapitel „Zylinderdetektion mithilfe der Hough-Transformation“(3) sich mit der theoretischen Grundlage der sequentiell zweiteiligen Hough-Transformation auf der Basis von [22] beschäftigt. Die eigentliche Entwicklung der randomisierten Hough-Transformation sowie die Erweiterung dieser etwa

durch einen Least-Square Ansatz, wird im Kapitel „Implementierung“(4) beschrieben, wobei die wichtigsten Hauptschritte des Algorithmus mithilfe von Pseudocode am Ende des passenden Abschnittes noch einmal verdeutlicht werden. In diesem Kapitel werden auch Schwächen des in [22] vorstellten Algorithmus aufgezeigt und wie der in dieser Arbeit vorgestellte Ansatz diese minimieren kann. Das Kapitel „Experimente und Auswertung“(5) befasst sich mit der Auswertung der Zylinderdetektion, wobei diese zuerst auf simulierten Daten validiert und anschließend auf realen Daten erneut getestet werden, um mögliche Schwächen des Algorithmus zu finden.

Kapitel 2

Stand der Technik

Aufgrund der vielfältigen Einsatzmöglichkeiten von Zylindern gibt es in der heutigen Zeit verschiedene Methoden um diese in 3D Punktwolken zu detektieren. Im Allgemeinen kann man die Zylindererkennungsmethoden in zwei Klassen unterteilen, wobei die erste im Vorfeld eine Segmentierung braucht und die zweite auf den rohen Punktwolken arbeitet.

2.1 Zylinderdetektion mit segmentierten Punktwolken

Bei der ersten Methode versucht man ein Zylindermodell auf eine Menge von segmentierten Punkten zu fitten. Dies erfolgt meistens über die Minimierung der orthogonalen Distanzen zur Zylinderoberfläche, wofür man etwa die Methode der kleinsten Quadrate (Least-Square Ansatz) nutzen kann [16]. Wichtig hierbei sind gute Initialisierungsparameter, um lokale Minima zu vermeiden. Ein Nachteil dieses Ansatzes ist die starke Abhängigkeit von der Qualität der Vorsegmentierung [24]. Gerade bei realen Daten kann die Qualität durch Rauschen oder Ausreißer negativ beeinflusst werden.

Ein solcher Ansatz kann geeignet sein, wenn aufgrund vieler Ausreißer keine automatische Detektion der Zylinder möglich ist und somit die Eingabe eines Nutzers für eine erfolgreiche Detektion benötigt wird. So wurde etwa für die Toolbox 3DTK [1] ein Ansatz implementiert, welcher durch das manuelle Selektieren einiger Zylinderpunkte aus der Punktwolke, mithilfe eines Least-Square Ansatzes einen bestmöglichen Zylinder in die Vorauswahl legt. Da für diesen Ansatz jedoch die Diskretisierung der Zylinderachse von Nöten ist und die selektierten Punkte meist fehlerbehaftet sind, muss diese erste Schätzung noch weiter verbessert werden. Hierfür werden zusätzlich die Punkte aus der nächsten Zylinderumgebung betrachtet. Mithilfe eines RANSAC Ansatzes und der bereits im Vorfeld berechneten Zylinderachse, ist es möglich aus den neuen Punkten eine verbesserte Zylinderachse zu bestimmen. Diese ermöglicht es anschließend den Radius des Zylinders mithilfe eines Least-Square Ansatzes noch einmal feiner zu bestimmen. Im Zuge dieser Implementierung wurde eine Anzeigefunktion (`qtshow`) von 3DTK [1] erweitert, sodass es ebenfalls möglich ist Zylinder in Punktwolken darzustellen. Diese Funktion wurde in

dieser Arbeit zum Beispiel in Kapitel 5 verwendet um die detektierten Zylinder darzustellen.

2.2 Zylinderdetektion auf unsortierten Punktwolken

Diese Abhängigkeit von einer Vorsegmentierung wird im zweiten Ansatz vermieden. Durch RANSAC oder Hough-Transformation basierte Methoden können Zylinder ohne vorherige Segmentierung aus Punktwolken extrahiert werden.

Ein RANSAC Ansatz nutzt ein zufallsbasiertes Samplingverfahren mit möglichst wenig Messpunkten aus, um ein Modell in die Datenpunkte zu legen. Es werden so lange weitere mögliche Modelle berechnet, bis eine ausreichend gute Lösung gefunden wird. Im Falle der Zylinderdetektion kann ein Zylindermodell bereits aus 2 Messpunkten sowie deren Oberflächennormalen berechnet werden [23]. Sind Bedingungen für die Ausrichtung der Zylinderachsen bekannt, kann zudem ein RANSAC Ansatz verwendet werden, um Zylinder in sehr großen Punktwolken zu detektieren [15]. RANSAC kann jedoch auch als eine Teillösung für die Zylindersuche verwendet werden. So kann etwa die Zylinderachsenorientierung mithilfe von RANSAC über die Gauß Sphäre berechnet werden und anschließend als Eingabe für ein Least-Square Fitting genutzt werden [8]. Der große Nachteil eines RANSAC basierten Ansatzes ist die Parameterwahl, welche stark von dem Signal-Rausch-Verhältnis abhängt. Diese Parameter sind mitunter für den Abbruch des Samplingverfahrens verantwortlich. Eine falsche Parameterwahl kann die Wahrscheinlichkeit für ein schlechtes Ergebnis erhöhen, was im schlimmsten Fall bedeuten kann, dass vorhandene Zylinder nicht gefunden werden.

Im Gegensatz zu RANSAC betrachtet eine Hough-Transformation alle Messpunkte und nutzt diese aus um die zugehörigen Zellen im Hough-Raum zu akkumulieren. Ein globales Maximum im Hough-Raum ist somit eine Lösung für ein mögliches Modell. Ein Hough-Transformations Ansatz hat den Vorteil stabil zu sein, wenn viele vorhandene Zylinderinstanzen und Ausreißer vorhanden sind. Ein großer Nachteil ist jedoch der Rechenaufwand, der mit der Anzahl der Eingabepunkte n , der Anzahl der Samplingpunkte einer Hough-Raum Dimension s , aber vor allem auch mit der Anzahl an gesuchten Parametern p zusammenhängt. Die Raum- beziehungsweise Zeitkomplexität eines Fittings im dreidimensionalen Raum kann approximiert werden mit $O(s^p)$ beziehungsweise $O(s^{p-1} \cdot n)$. Da ein Zylinder 5 Freiheitsgrade besitzt, kann eine Hough-Transformation sehr schnell die Grenzen des gewünschten Rechen- und somit auch Zeitaufwandes sprengen. Der Artikel [22] liefert hierfür die Grundidee einer zweiteiligen Hough-Transformation zur Detektion von Zylindern. Diese Idee dient als Grundlage der in dieser Arbeit entwickelten Zylinderdetektion.

Im folgenden Kapitel „Zylinderdetektion mithilfe der Hough-Transformation“ (3) wird deshalb noch einmal genauer auf die zugrundeliegende Theorie dieser Masterarbeit, die Hough-Transformation, die mögliche Aufteilung in Achsenorientierung sowie Position und Radiusberechnung und anschließend auf eine mögliche Randomisierung des Ansatzes eingegangen.

Kapitel 3

Zylinderdetektion mithilfe der Hough-Transformation

3.1 Hough-Transformation

Die Hough-Transformation ist eine Methode zur Detektion von parametrisierten Objekten. Sie wird häufig im 2-dimensionalen Bereich für Linien und Kreisdetektion verwendet. Doch sie kann auch zur Ebenendetektion im 3-dimensionalen Raum oder wie in diesem Fall zur Zylinderdetektion verwendet werden.

Ein Nachteil der Standard Hough-Transformation ist die mit jedem Parameter stark ansteigende Raum- beziehungsweise Zeitkomplexität. Ein Zylinder besitzt 5 Freiheitsgrade und wird im Folgenden durch 5 Parameter dargestellt. Er sollte daher nicht durch eine einzige Hough-Transformation detektiert werden. Stattdessen ist es sinnvoll zwei sequentielle Hough-Transformationen zu verwenden, wobei die erste 2 Parameter für die Zylinderachse und die zweite 3 Parameter für Radius und Position des Zylinders bestimmt. [22]

Da die 2D Hough-Transformation zur Zylinderachsendetektion aufgrund der Verwendung von sphärischen Normalen als Eingabe bereits eine höhere Komplexität aufweist, wird im folgenden Abschnitt kurz das Grundprinzip einer 2D Hough-Transformation anhand des Standardbeispiels der Liniendetektion erklärt. Fachkundigen Lesern, denen die Grundlagen der Hough-Transformation bekannt sind, wird aus diesem Grund empfohlen den Abschnitt „2D Hough-Transformation“(3.2) zu überspringen und in dem Abschnitt „Zylinderdetektion“(3.3) fortzufahren.

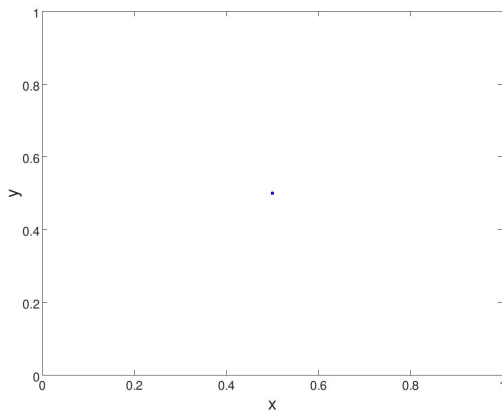
3.2 2D Hough-Transformation

Die grundlegende Idee der Hough-Transformation kann gut anhand der Detektion kollinear Punkte beschrieben werden. Als Eingabe wird in diesem Beispiel ein unsortiertes 2D Punkteset \mathbf{P} genutzt, wobei jeder Punkt \mathbf{p}_i einen kartesischen x -, sowie y -Wert besitzt. Für ein möglichst einfaches Beispiel wird angenommen, dass die Punkte auf einer Linie \mathbf{l}_0 liegen oder frei im Raum verteilt sind.

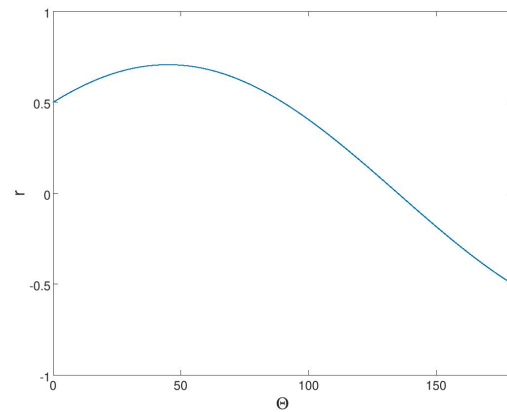
Sollen die kollinearen Punkte mithilfe einer Hough-Transformation detektiert werden, so muss zuerst die mathematische Beschreibung einer Geraden definiert werden. Im kartesischen Koordinatensystem wird eine Gerade häufig über ihre Steigung m und ihren y -Achsenabschnitt t beschrieben: $y = mx + t$. Ein Nachteil dieser Schreibweise ist die Darstellung von Geraden parallel zur y -Achse, da in diesem Fall die Steigung unendlich ist. Um diese Linien trotzdem finden zu können, ist für die Hough-Transformation eine andere Darstellung interessanter, die sogenannte Hessesche Normalform:

$$r = x \cos \Theta + y \sin \Theta \quad . \quad (3.1)$$

Hierbei wird die Gerade durch ihren senkrechten Abstand zum Ursprung r und Θ den Winkel zwischen r und der x -Achse beschrieben, somit liegt Θ im Intervall $[0, \pi)$. Daraus folgt, dass eine Gerade in einem Koordinatensystem mit den Koordinaten r und Θ als ein Punkt (r_0, Θ_0) dargestellt wird.



(a) Punkt \mathbf{p}_0



(b) Alle Linien durch \mathbf{p}_0 im (r, Θ) -Raum

Abbildung 3.1: Darstellung eines Punktes \mathbf{p}_0 im (r, Θ) -Raum

Zu Beginn der Detektion ist diese Gerade und somit Θ_0 und r_0 jedoch noch unbekannt, stattdessen sind nur die kartesischen Koordinaten der Punkte in \mathbf{P} bekannt. Die Grundidee der Hough-Transformation ist dabei, dass ein kartesischer Eingabepunkt \mathbf{p}_i alle Geraden darstellt die durch \mathbf{p}_i gehen können. Dies führt dazu, dass ein beliebiger Punkt \mathbf{p}_i als eine Cosinus

Funktion im (r, Θ) -Raum dargestellt wird (siehe Abb. 3.1), die alle möglichen Geraden durch \mathbf{p}_i beschreibt.

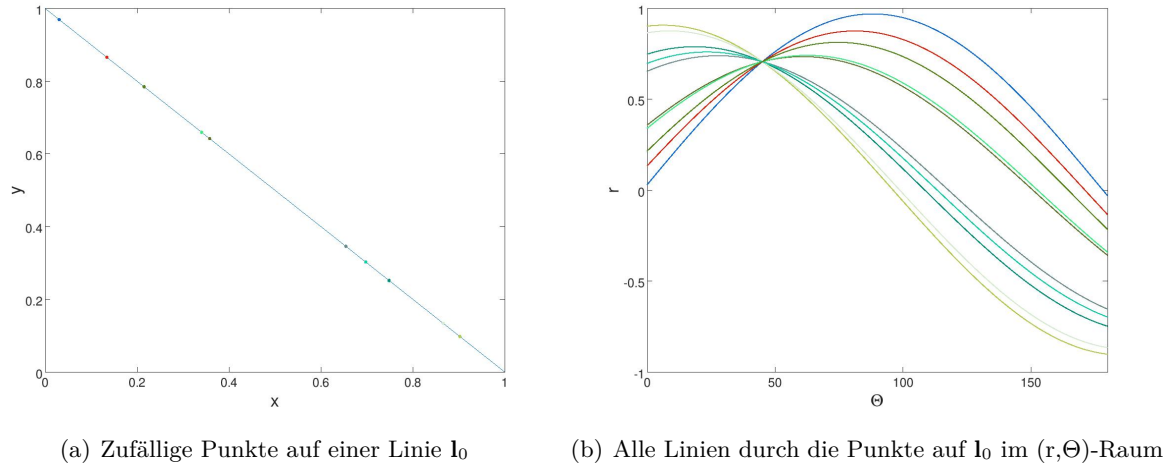
(a) Zufällige Punkte auf einer Linie l_0 (b) Alle Linien durch die Punkte auf l_0 im (r, Θ) -Raum

Abbildung 3.2: Darstellung mehrerer Punkte auf der Linie l_0 im (r, Θ) -Raum

Die Hough-Transformation nutzt diese Darstellung aus, um Geraden zu detektieren. Für eine schöne bildliche Darstellung wird angenommen, dass l_0 die Steigung $m_0 = -1$ und den y-Achsenabschnitt $t_0 = 1$ besitzt oder dementsprechend $\Theta_0 = 45^\circ$ und $r_0 \approx 0.7$ gilt. Es befinden sich im x-Achsenbereich $[0,1]$ zehn beliebige Punkte \mathbf{p}_i auf der Gerade l_0 , mit $i = [0, 9]$. Stellt man nun für jeden Punkt \mathbf{p}_i alle Linien im (r, Θ) -Raum dar, so erhält man zehn Cosinusfunktionen, die sich alle im Punkt $\Theta = 45^\circ$ und $r \approx 0.7$ schneiden, da dies die einzige mögliche Linie ist, die durch alle Punkte \mathbf{p}_i gehen kann (siehe Abb. 3.2).

In der Praxis wird der (r, Θ) -Raum quantisiert, um den oder die Schnittpunkt(e) zu finden. Diese Quantisierung teilt den Raum in kleine Teilabschnitte auf, ähnlich wie die einzelnen Pixel in einem digitalen Bild. Die aufgeteilten Abschnitte dienen dabei der Hough-Transformation als Akkumulatorzellen, weswegen der quantisierte (r, Θ) -Raum auch als Akkumulator bezeichnet wird. Jede Zelle besitzt zu Beginn der Hough-Transformation den Wert 0. Die Standard Hough-Transformation berechnet für jeden beliebigen Eingabepunkt p_i , alle möglichen (r_i, Θ_i) -Paare und erhöht die passenden Zellen des Akkumulators um eins. Der Endwert einer beliebigen Zelle z_i gibt an, wie viele Punkte für die passende Gerade l_i gestimmt haben. Ist der Endwert von z_i größer als ein zuvor bestimmter Schwellwert, so wurde eine Gerade detektiert. [11]

3.3 Zylinderdetektion

Die folgenden Abschnitte beschäftigen sich mit der Theorie der sequentiellen zwei schrittigen Hough-Transformation zur Zylinderdetektion wie sie in [22] vorgestellt wird. Hierfür wird zuerst die Parametrisierung der Zylinder beschrieben, um anschließend beschreiben zu können

wie durch geschicktes Aufteilen eine Zylinderdetektion mit einer zweidimensionalen und einer weiteren dreidimensionalen Hough-Transformation möglich ist.

3.3.1 Zylinder Parametrisierung

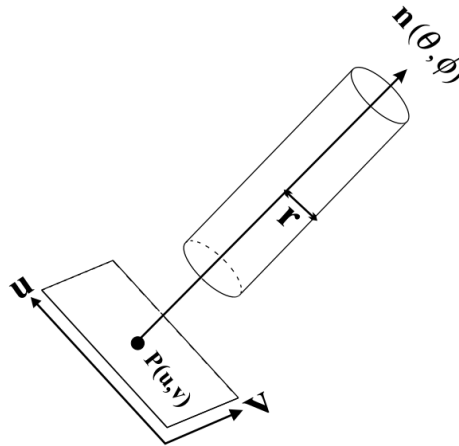


Abbildung 3.3: Zylinder Parametrisierung aus [22]

Ein Zylinder besitzt insgesamt fünf Freiheitsgrade, die man auf verschiedene Arten parametrisieren kann. Wichtig für die Parametrisierung ist, dass die Hough-Transformation mit einer großen Anzahl an Parametern sehr schnell unhandlich wird. Daher muss man die Parameter so wählen, dass man die Zylindersuche in zwei sequentielle Schritte unterteilen kann. Diese Unterteilung hat den Vorteil, die steigende Raum- und somit Zeitkomplexität zumindest teilweise zu umgehen.

Im Falle eines Zylinders kann man die Suche in zwei sequentielle Schritte unterteilen. Im ersten Schritt kann man mithilfe einer 2D Hough-Transformation eine starke Hypothese für die Zylinderachsen berechnen. Im zweiten Schritt wird anschließend eine 3D Hough-Transformation genutzt, um für jede im 1. Schritt gefundene Achse einen Radius sowie die Position zu bestimmen. Durch diese sequentielle Aufteilung wird die maximale Anzahl an Parametern einer Hough-Transformation auf drei reduziert. [22]

Um diese Aufteilung zu ermöglichen wird der Zylinder mit 5 Parametern beschrieben. Für die erste 2D-Hough-Transformation wird die Richtung der Zylinderachse vom Ursprung mithilfe der Polarkoordinaten φ und Θ beschrieben. sie ist definiert als:

$$\mathbf{n} = [\cos(\Theta) \sin(\varphi), \sin(\Theta) \sin(\varphi), \cos(\Theta)] \quad . \quad (3.2)$$

Um anschließend im zweiten Schritt die Position sowie den Radius zu bestimmen, wird der Zylinder entlang seiner Zylinderachse auf eine Ebene projiziert, wobei das Ebenenkoordinatensystem

mit den Achsen \mathbf{u} und \mathbf{v} orthonormal zur Zylinderachse ist. Somit wird aus dem dreidimensionalen Zylinder, ein zweidimensionaler Kreis mit dem Mittelpunkt $\mathbf{c}_m = (u_m, v_m)$ und dem Radius r . Für die Kreisgleichung gilt:

$$r^2 = (x_p - u_m)^2 + (y_p - v_m)^2 \quad . \quad (3.3)$$

Wobei (x_p, y_p) das projizierte unsortierte Punkteset auf dem Kreisrand beschreibt. In der Realität sind (x_p, y_p) jedoch fehlerbehaftet. Daraus folgen die parametrischen Gleichungen des Kreises:

$$\begin{aligned} x_p &= u_m + r \cos(\Theta_p) \\ y_p &= v_m + r \sin(\Theta_p) \end{aligned} \quad . \quad (3.4)$$

Im folgenden wird diese Parametrisierung verwendet, um die Theorie hinter einer zweistufigen Hough-Transformation zur Zylinderdetektion zu erläutern, wobei als Grundlage stets die Publikation [22] verwendet wird.

3.3.2 Zylinderachsendetektion durch Hough-Transformation

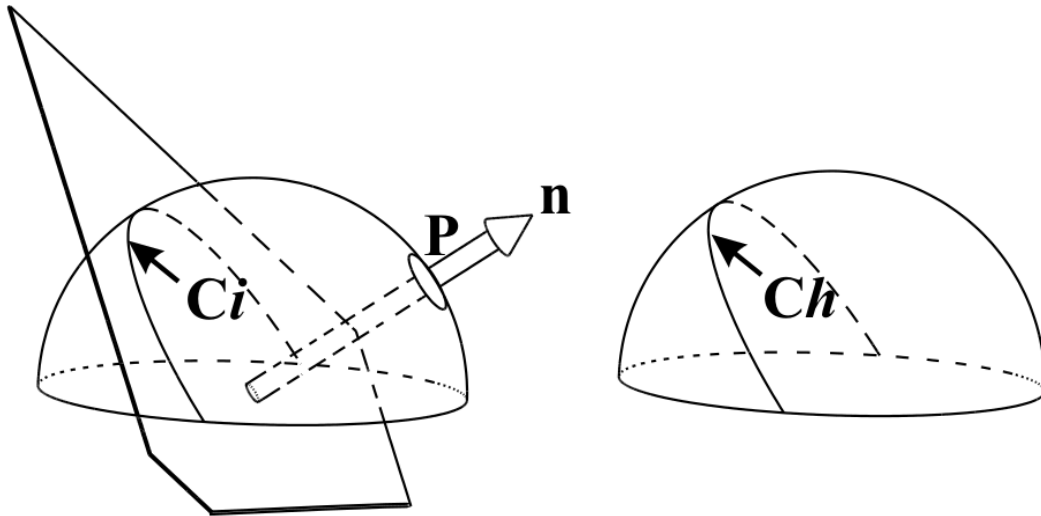
Für eine erfolgreiche Zylinderdetektion muss im ersten Schritt eine starke Hypothese für die Zylinderachsen gefunden werden.

Eine Hough-Transformation kann sich hierbei die Besonderheit zu nutze machen, dass die Normalen eines Zylinders einen Großkreis auf der Gaußschen Sphäre bilden. Dieser Großkreis bildet sich durch das Schneiden der Gaußschen Sphäre mit einer Ursprungsebene, deren Normalenvektor gleich der Zylinderachse ist. Somit detektiert die Hough-Transformation im Grunde genommen eine Ursprungsebene, welche die Einheitskugel schneidet.

Die Standard Hough-Transformation um eine Ebene zu finden, benötigt zur Ebenendetektion einen dreidimensionalen Hough-Raum, wobei zwei Parameter die Normale der Ebene in sphärischen Koordinaten beschreiben und ein weiterer die Distanz zum Ursprung parametrisiert. Da es sich bei der gesuchten Schnittebene immer um eine Ursprungsebene handelt, ist die Distanz hierbei stets Null. Somit besitzt die Hough-Transformation zur Detektion einer Ursprungsebene nur noch zwei Parameter. Des Weiteren wird eine Ebene gesucht, die die Gaußsche Sphäre schneidet. Das bedeutet, dass jeder Eingabepunkt für eine Kreisregion im Hough-Raum votet. [22]

In Abbildung 3.4 wird das Votingverfahren für den Punkt \mathbf{P} beispielhaft dargestellt. In der linken Abbildung wird die Normale \mathbf{n} vom Ursprung zu \mathbf{P} dargestellt. Die ebenfalls eingezeichnete Ursprungsebene wird durch den Koordinatenursprung und \mathbf{n} definiert. Der Schnitt der Ursprungsebene und der Halbkugel wird als \mathbf{Ci} markiert. In der Abbildung rechts wird noch einmal der Großkreis \mathbf{Ch} dargestellt, für den \mathbf{P} mithilfe des berechneten \mathbf{Ci} votet.

Um ein solches Voting zu implementieren, muss zuerst für jeden Eingabepunkt seine Normale auf der Einheitskugel berechnet werden. Dies führt dazu, dass als Eingabe eine aus den Normalen der Punktwolke berechnete Gaußsche Sphäre verwendet wird. Da aber auch der Hough-Raum



(a) Punkt \mathbf{P} auf der Gaußschen Eingabesphäre [22] (b) \mathbf{P} votet für dargestellten Großkreis [22]

Abbildung 3.4: Darstellung des Votingverfahrens für Punkt \mathbf{P} (Abb. aus [22])

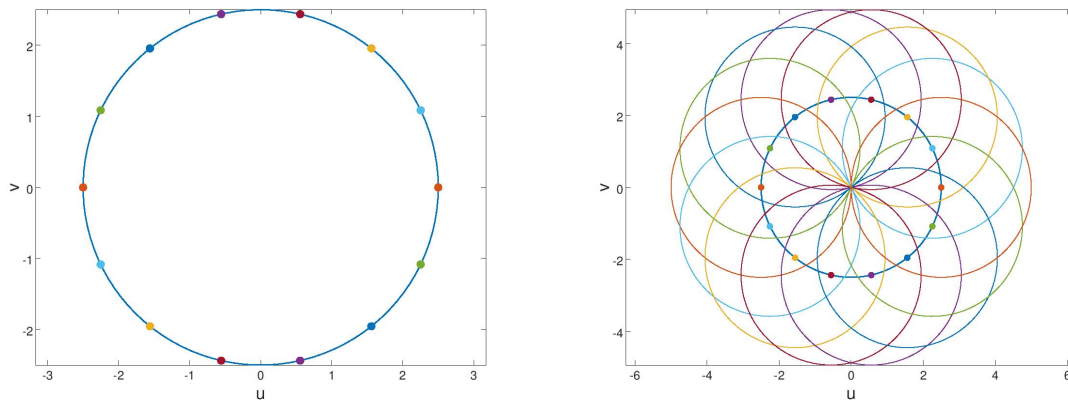
nur aus Orientierungen der Ebenennormalen besteht, kann man diesen ebenfalls als eine Gaußsche Sphäre interpretieren. Um Verwechslung zu vermeiden werden im Folgenden deswegen die Begriffe Gaußsche Eingabesphäre (berechnet aus der Punktwolke) oder Gaußsche Hough Sphäre (die den Hough-Raum darstellt) genutzt.

Jede Normale, beziehungsweise der dazugehörige Punkt auf der Gaußschen Eingabesphäre, votet dabei für alle orthogonalen Ebenen auf der Gaußschen Hough Sphäre. Jede Richtung besitzt somit ein orthogonales Set an dreidimensionalen Punkten die eine Ebene formen. Das besondere dabei ist, dass die Magnitude jedes einzelnen Setpunktes 1 ist. Dieser Zusammenhang garantiert, dass für jede Schnittebene mit der Gaußschen Hough Sphäre ein Großkreis entsteht. Im Grunde genommen votet also jede Eingabenormale für den großen Kreis im Hough-Raum, der denselben Normalenvektor besitzt wie der Punkt. [22]

Der Schnittpunkt aller Großkreise eines Zylinders bildet ein Maximum im Hough-Raum und schätzt die Zylinderachsenrichtung ab.

3.3.3 3D Hough-Transformation

Die 2D-Hough-Transformation lässt sich jedoch auch auf parametrisierte dreidimensionale Objekte wie Kreise erweitern. Kern der Hough-Transformation bleibt hierbei stets der Akkumulator und die Detektion seiner Maxima. Es ist jedoch wichtig, die Auswirkung dieser Erweiterung auf die Komplexität zu verstehen. Bei einem dreidimensionalen Fit kann man die Raumkomplexität approximiert mit $O(s^p)$ beschreiben, wobei s die Anzahl der Samples entlang einer Houghdimension beschreibt und p die Anzahl der gesuchten Parameter ist. Für die Zeitkomplexität gilt dementsprechend $O(s^{p-1}n)$, wobei n die Anzahl der Punkte ist. Da es



(a) Kreis \mathbf{K} mit Radius 2,5 und einem Punktset \mathbf{P}_k (b) Kreis \mathbf{K} mit Punktset \mathbf{P}_k , wobei um jeden Punkt auf dem Kreisrand mit der Größe $n_k = 15$ aus \mathbf{P}_k ein neuer Kreis mit Radius 2,5 gezeichnet ist

Abbildung 3.5: Darstellung der Grundidee, der Standard Hough-Transformation zur Kreisdetektion

sich in diesem speziellen Fall, um die Auswertung von 3D-Punktwolken eines Laserscanners handelt, kann n in der Größenordnung von Millionen Punkten liegen. Damit wird die Hough-Transformation für mehr als drei Parameter sehr schnell rechenintensiv. [22]

3.3.4 Positions- und Radiusberechnung

Die zuvor bestimmte Hypothese für die Zylinderachsenrichtung kann genutzt werden, um die Position sowie den Radius des Zylinders zu berechnen. Werden mehrere Zylinderachsen im Schritt 1 gefunden, so muss die Position- beziehungsweise Radiusberechnung für jede Achse wiederholt werden.

Zu Beginn der Positions- beziehungsweise Radiusbestimmung für eine beliebige detektierte Achse \mathbf{n} , müssen alle Eingabepunkte entlang der Zylinderachsenrichtung auf eine Ebene projiziert werden. Ist die Achsenrichtung perfekt, so wird der Zylinder als exakter Kreis auf der Ebene dargestellt. Um die Punkte möglichst schnell Projizieren zu können, wird ein orthonormales Koordinatensystem berechnet, dessen eine Achse gleich der Zylinderachsenorientierung \mathbf{n} ist. Insgesamt gibt es zwei Möglichkeiten dieses System mithilfe des Zylinderachsenvektors aufzustellen: Hierfür kann die Gram-Schmidt Orthonormalisierung oder eine SVD (Singular Value Decomposition) verwendet werden. Unabhängig welche Technik genutzt wird, das Ergebnis ist ein orthonormales Koordinatensystem beschrieben durch die drei Basisvektoren $(\mathbf{u} \ \mathbf{v} \ \mathbf{n})$, wobei \mathbf{n} die Zylinderachse ist. Diese Basisvektoren können anschließend verwendet werden um die Punkte auf die Ebene zu projizieren. [22]

Sobald für jeden 3D-Eingabepunkt seine zweidimensionale Projektion berechnet wurde, können mithilfe eines Kreisfitting-Algorithmus die Kreise der projizierten Zylinder mit Achse \mathbf{n} detektiert werden. Hierbei schlägt [22] vor eine Hough Transformation nach [13] zu implementieren.

Die zu Grunde liegende Idee ist dabei relativ einfach: Auf einem Kreis \mathbf{K} mit dem Radius r_k liegt ein n_k großes Set von Punkten \mathbf{P}_k . Werden nun n_k neue Kreise \mathbf{K}_n mit dem Radius r_k erstellt, wobei die Mittelpunkte jeweils einem Punkt aus dem Set \mathbf{P}_k entsprechen, so schneiden sich alle neuen Kreise \mathbf{K}_n im Mittelpunkt von \mathbf{K} . Dies wird noch einmal durch die Darstellung 3.5 verdeutlicht. Im linken Bild ist ein Kreis \mathbf{K} mit einem beispielhaften Radius von 2,5 zu sehen. Auf seinem Rand liegen 15 Punkte. Jeder dieser Punkte dient in der rechten Abbildung als der Mittelpunkt eines neuen Kreises mit ebenfalls den Radius 2,5. Es ist zu erkennen, dass sich alle neuen Kreise in dem ursprünglichen Kreismittelpunkt von \mathbf{K} schneiden. Diese Gegebenheit wird für die Hough-Transformation genutzt. Für einen beliebigen gegebenen Radius r votet ein Punkt im Hough-Raum stets für eine Kreisregion mit sich selbst als Mittelpunkt. So würde ein beliebiger Punkt mit den projizierten Koordinaten (u_p, v_p) im Hough-Raum für alle Zellen voten, die beschrieben werden durch:

$$(r \cos \omega + u_p, r \sin \omega + v_p) \quad . \quad (3.5)$$

Das Maximum im Hough-Raum beschreibt somit die Orientierung und den Radius des Zylinders direkt. [22]

Ein großer Nachteil dieser Methode ist jedoch, dass in den meisten Fällen der Radius für die Zylinder nicht bekannt ist. Somit muss r in diesem Verfahren gesampelt werden. Wenn kein Maximum oder Minimum für r gegeben und die benötigte Auflösung unbekannt ist, fällt dies schwer. Zudem steigt mit jedem weiteren gesampelten Radius die Zeitkomplexität an. Aus diesem Grund wurde eine randomisierte Hough-Transformation kombiniert mit einem Least-Square Ansatz verwendet. Im folgenden Abschnitt 3.3.5 werden die allgemeinen Vorteile einer randomisierten Hough-Transformation im Vergleich zur Standard Hough-Transformation beschrieben und im Kapitel „Implementierung“ (Abschnitt 4.2) wird die Entwicklung der randomisierten Kreisdetektion erklärt.

Generell kann für die Kreisdetektion jeder bekannte Algorithmus verwendet werden, solange am Ende alle projizierten Zylinder mit der Achse \mathbf{n} gefunden werden. Anschließend muss nur noch der Kreismittelpunkt zurück in das Weltkoordinatensystem transformiert werden. So erhält man einen 3D Punkt auf der Zylinderachse und ermittelt somit die Position des Zylinders.

3.3.5 Randomisierte Hough-Transformation

Trotz der sequentiellen zweiteiligen Aufteilung kann die Berechnung einer Standard Hough-Transformation sehr rechenaufwendig werden, da die Größe des Eingabedatensets mehrere hunderttausend Punkte umfassen kann. Mithilfe eines geschickt gewählten Akkumulators kann man für die erste Hough-Transformation den Hough-Raum gut mithilfe einer quantisierten halben Einheitskugel darstellen (siehe für eine genauere Beschreibung Abschnitt 4.1.2). Doch beim zweiten Schritt wird dies bereits schwerer, da für die Kreissuche hinsichtlich des Radius und der Position des Zylinders aufgrund der Allgemeingültigkeit des Ansatzes keine einfachen Grenzen gesetzt sind. Somit ist es für die Kreisdetektion schwierig einen geeigneten Hough-Raum mit einer ausreichend großen Auflösung im Vorfeld aufzubauen, da die Grenzen des Hough-Raums, sowie die interessanten Bereiche, nur schwer greifbar sind.

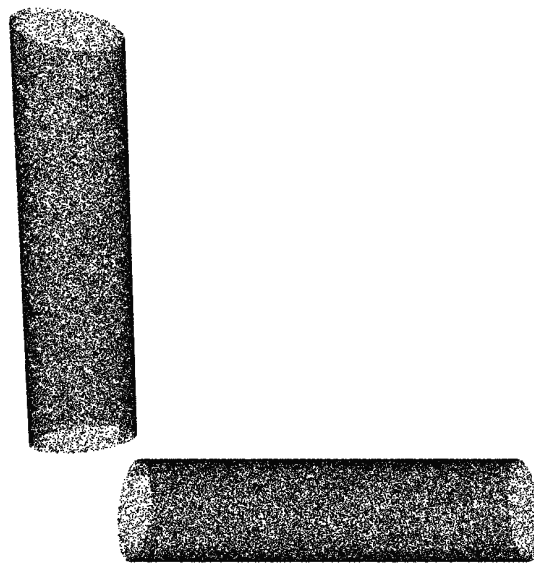
Die Anzahl der nötigen Eingabepunkte, sowie das Problem der Diskretisierung des Hough-Raums lassen sich mithilfe einer randomisierten Hough-Transformation (RHT) verringern. Ziel der RHT ist es nur eine Teilmenge der Eingabepunkte anzusehen, um mit einer gewissen Wahrscheinlichkeit das parametrisierte Objekt zu bestimmen. Die RHT nutzt einen Akkumulator, der die Zellen passend zu den Treffern expandiert, anstatt im Vorfeld den Hough-Raum zu diskretisieren. Dies hilft etwa im Falle der Kreisdetektion, bei der der Akkumulator nur dort expandiert wird, wo sich mit einer gewissen Wahrscheinlichkeit ein Kreis befindet. [10]

Das Vorgehen der RHT ist für die zwei Schritte der Zylinderdetektion sehr ähnlich. Anstatt eines Punktes wählt die RHT genug zufällige Punkte aus, um das parametrisierte Objekt zu berechnen. Im Falle einer Ursprungsebene sind dies mindestens zwei und im Falle eines Kreises mindestens drei Punkte. Anschließend wird das berechnete Objekt genutzt, um den dazu passenden Wert im Hough-Raum zu akkumulieren. Das bedeutet etwa für die Zylinderachsendetektion, dass zwei zufällig gewählte sphärische Normalenpunkte für eine Zelle im Akkumulator voten, während ein sphärischer Normalenpunkt bei der Standard Hough-Transformation für einen Großkreisabschnitt im Hough-Raum votet. Dieses Verfahren ermöglicht es nur einen Bruchteil der Eingabepunkte zu betrachten, solange der Signal-Rausch Abstand groß genug ist. Im Optimalfall wird durch diese Methode der Hough-Raum zudem nur zum Teil betrachtet.

Ein Nachteil der RHT ist die Abhängigkeit der Parameter vom Signal-Rausch Abstand. Wie häufig etwa zufällig n Punkte gewählt und daraus das parametrisierte Objekt bestimmt werden muss, hängt in dem Spezialfall der Zylinderdetektion stark davon ab, wie wahrscheinlich es ist zwei beziehungsweise drei Punkte derselben Zylinderachsenrichtung beziehungsweise desselben Kreises zufällig auszuwählen. So geht neben der Anzahl der Ausreißer in der Kreisdetektion etwa auch mit ein, wie viele Zylinder dieselbe Achsenrichtung besitzen, da diese alle in derselben Kreisdetektion gefunden werden. Das bedeutet es gibt keine universalen Parameter, die für jede Punktwolke genutzt werden können, sondern die Parameter sind stark von der Eingabe abhängig und müssen somit vom Nutzer bestimmt werden. [10]

Kapitel 4

Implementierung



created by 3DJK
<http://3dreedk.de>

Abbildung 4.1: Punktwolke von zwei senkrecht zueinander stehende simulierte Zylinder

Dieses Kapitel beschäftigt sich mit der Implementierung der Zylinderdetektion. Dabei wird die Theorie aus dem Kapitel „Zylinderdetektion mithilfe der Hough-Transformation“ (siehe Abb. 3) angewendet, erweitert und teilweise angepasst, um einige Spezialfälle besser abzudecken und eine effizientere Implementierung zu ermöglichen. Wichtige Teilschritte werden am Ende eines Abschnittes mithilfe von Pseudocode noch einmal zusammengefasst und verdeutlicht.

4.1 Zylinderachsendetektion

Zum besseren Verständnis werden die einzelnen Teilschritte der Zylinderachsendetektion anhand eines einfachen generierten Beispiels dargestellt. Hierfür wurden zwei senkrecht zueinander stehende Zylinder mit dem Radius 2,5 cm und der Länge 20cm gewählt (dargestellt in Abbildung 4.1). Jeder Zylinder wurde mit 40000 Punkten gesampelt. Für eine möglichst gute Darstellung der Theorie wurde in diesem Fall kein simuliertes Rauschen verwendet.

4.1.1 Vorbereitung - Berechnung sphärischer Normalen

Als Eingabe für die Zylinderachsendetektion wird ein unsortiertes Punkteset \mathbf{S} verwendet. Diese Punkte an sich können jedoch nicht sofort dem Akkumulator übergeben werden, da im Falle der Zylinderachsendetektion mithilfe der sphärischen Normalen eine Ursprungsebenendetektion berechnet wird. Aus diesem Grund muss eine Vorverarbeitung des Punktesets \mathbf{S} stattfinden.

Als erster Schritt wird für jeden Punkt $\mathbf{p} = (p_x, p_y, p_z)$ aus dem Set \mathbf{S} eine Normale berechnet. Hierfür schlägt [22] vor die Methode von Hoppe zu verwenden. Diese ermöglicht es mithilfe der Methode der kleinsten Quadrate auch auf rohen Punktwolken Normalen zu berechnen. Die Grundidee ist dabei eine Ebene bestmöglich auf den Punkt und seine nächste Nachbarschaft zu legen, um anschließend einen auf der Ebene senkrecht stehenden Vektor zu bestimmen. Die bestmögliche Ebene wird hierbei durch die Minimierung der Fehlerquadrate der Punkteabstände bestimmt. [12]

Normalen Berechnung

Gegeben sei ein Punkt \mathbf{p} aus dem Set \mathbf{S} , mit den Koordinaten $(x, y, z)^T$. Es werden für \mathbf{p} k nächste Nachbarn $(\mathbf{p}'_{1\dots k})$ ausgewählt. Diese werden nicht wie in [22] direkt über die unsortierte Punktwolke, sondern mithilfe eines aus den Punkten erstellten Rangeimages bestimmt. Gesucht ist eine Tangentenebene $\mathbf{T}\mathbf{p}(\mathbf{p})$, die durch den auf ihr liegenden Punkt \mathbf{p} , sowie einen Einheitsnormalenvektor $\mathbf{n} = (n_x, n_y, n_z)$, mit $|\mathbf{n}| = 1$, definiert ist. Es gilt also:

$$n_x x + n_y y + n_z z - d = 0 \quad . \quad (4.1)$$

Die Methode der kleinsten Quadrate bestimmt $\mathbf{T}\mathbf{p}(\mathbf{p})$, durch die Minimierung des Fehlers:

$$e = \sum_{i=1}^k (\mathbf{p}'_i \mathbf{n}^T - d)^2 \quad . \quad (4.2)$$

Die Lösung des minimalen Fehlers kann mithilfe der Kovarianzmatrix bestimmt werden:

$$\mathbf{M} = \frac{1}{k} \sum_{i=1}^k (\mathbf{p}'_i - \bar{\mathbf{p}})(\mathbf{p}'_i - \bar{\mathbf{p}})^T, \text{ wobei } \bar{\mathbf{p}} = \frac{1}{k} \sum_{i=1}^k \mathbf{p}'_i \quad . \quad (4.3)$$

Der kleinste Eigenvektor der Matrix \mathbf{M} entspricht dabei \mathbf{n} . [12]

Ein Nachteil dieser Methode ist jedoch, dass die Berechnung der Matrix \mathbf{M} und ihres kleinsten Eigenvektors vergleichsweise rechenaufwendig ist und für jeden Eingabepunkt ausgeführt werden muss. Zudem ist der traditionelle Ansatz von k abhängig.

Eine Möglichkeit die Normalen ohne die Abhängigkeit von k zu Berechnen kann etwa durch den normalisierten Ansatz der kleinsten Quadrate erreicht werden. Die Idee dieses Ansatzes ist, dass Rauschen nicht wie im traditionellen Ansatz als unabhängig und identisch verteilt betrachtet wird, sondern das Rauschen größtenteils durch einen Entfernungsmessfehler zustande kommt, welcher sich linear zur Sichtrichtung ausbreitet. Anstatt also direkt die Eigenvektoren aus \mathbf{M} zu bestimmen, ist es bei einem solchen Fehler sinnvoller die Koordinaten der 3D-Eingabepunkte zu normalisieren, sodass die drei Hauptachsen von \mathbf{M} gleich der Einheitslänge sind. Anschließend wird die Matrix \mathbf{M} mithilfe der Cholesky Faktorisierung normalisiert:

$$\mathbf{M}' = \mathbf{K}^{-1}\mathbf{M}\mathbf{K}^T \quad . \quad (4.4)$$

\mathbf{K} ist dabei eine untere Dreiecksmatrix, für die gilt:

$$\sum_{i=1}^k \mathbf{p}_i \mathbf{p}_i^T = \mathbf{K}\mathbf{K}^T \quad . \quad (4.5)$$

Anschließend wird die Lösung durch den kleinsten Eigenwert von \mathbf{M}' berechnet. Die Cholesky Faktorisierung ist dabei nicht abhängig von der Anzahl der Nachbarn k , jedoch geht dies mit einem signifikant größeren Rechenaufwand einher. [6]

Mithilfe einer Approximation kann die Normalenberechnung beschleunigt werden, wobei dieser Ansatz natürlich weniger genau ist als der traditionelle Ansatz. Hierfür muss zuerst die Fehlerfunktion durch d^2 geteilt werden:

$$e = \sum_{i=1}^k (\mathbf{p}_i^T \tilde{\mathbf{n}} - 1)^2 \quad , \quad (4.6)$$

wobei $\tilde{\mathbf{n}}$ der gesuchte Normalenvektor der Ebene multipliziert mit einem Skalierungsfaktor ist. Die Division von d^2 führt jedoch für $d = 0$, also bei allen Ursprungsebenen zu einer Degeneration. Aus diesem Grund werden die k nächsten Nachbarn auch nicht direkt mit der Punktwolke gesucht, sondern mithilfe eines aus der Punktwolke erstellten Rangeimages. Dies hat den Vorteil, dass d niemals 0 ist. Somit gilt für $\tilde{\mathbf{n}}$:

$$\tilde{\mathbf{n}} = \tilde{\mathbf{M}}^{-1} \tilde{\mathbf{b}} \quad , \quad (4.7)$$

wobei $\tilde{\mathbf{M}} = \sum_{i=1}^k \mathbf{p}_i \mathbf{p}_i^T$ und $\tilde{\mathbf{b}} = \sum_{i=1}^k \mathbf{p}_i$. Anschließend kann \mathbf{n} durch die Normalisierung von $\tilde{\mathbf{n}}$ berechnet werden: $\mathbf{n} = \tilde{\mathbf{n}}/|\tilde{\mathbf{n}}|$. Diese Schreibweise kann, auch wenn in der Theorie nicht optimal, in der Praxis Vorteile mit sich bringen, da sowohl $\tilde{\mathbf{M}}$ als auch $\tilde{\mathbf{b}}$ effizient berechnet werden können. Mithilfe einer Approximation ist es jedoch möglich die Fehlerfunktion so zu vereinfachen, dass die Matrix $\tilde{\mathbf{M}}^{-1}$ nicht jedes Mal berechnet werden muss. Mithilfe einer Erweiterung von r_i^2 auf der rechten Fehlergleichungsseite ergibt sich:

$$e = \sum_{i=1}^k r_i^2 (\mathbf{v}_i^T \tilde{\mathbf{n}} - r_i^{-1})^2 \quad (4.8)$$

mit:

$$\mathbf{v}_i = \begin{pmatrix} \cos \theta_i \sin \phi_i \\ \sin \theta_i \sin \phi_i \\ \cos \theta_i \end{pmatrix} . \quad (4.9)$$

Damit ergibt sich für die Lösung:

$$\hat{\mathbf{n}} = \hat{\mathbf{M}}^{-1} \hat{\mathbf{b}} , \quad (4.10)$$

wobei $\hat{\mathbf{M}} = \sum_{i=1}^k \mathbf{v}_i \mathbf{v}_i^T$ und $\hat{\mathbf{b}} = \sum_{i=1}^k \mathbf{v}_i / r_i$. Das besondere an dieser approximierten Form ist, dass die Matrix $\hat{\mathbf{M}}^{-1}$ nur abhängig von den Bildparametern des Rangeimages ist und so vorberechnet werden kann. [6]

Neben der Methode der kleinsten Quadrate, die ursprünglich in [22] verwendet wird, kann man durch das Erstellen eines Rangeimages auch einen „Spherical Range Images“ (SRI) Differentialansatz verwenden. Beide Ansätze nutzen die Struktur der Rangeimages aus, um die Oberflächennormalen schneller zu bestimmen, als über die Rohdaten direkt.

Sollte die Position beziehungsweise die Positionen des Scanners im Raum nicht bekannt sein, so ist es nach der erfolgreichen Berechnung nur schwer möglich die Oberflächennormalen auszurichten. Dies führt dazu, dass die Normalen zwar senkrecht auf der Zylinderoberfläche stehen, allerdings vom Zylinder weg sowie in ihn hinein zeigen können. Eine Ausrichtung der Normalen ist für eine erfolgreiche Zylinderdetektion jedoch nicht nötig, dies wird in den folgenden Abschnitten 4.1.1 und 4.1.3 noch einmal genauer erklärt.

Transformation auf die Einheitssphäre

Nachdem für jeden Punkt \mathbf{p} eine Normale \mathbf{n} bestimmt wurde, können diese genutzt werden um die Rotationsmatrix \mathbf{R} zu berechnen, welche anschließend die Eingabe Normalen der Hough Transformation auf die Einheitssphäre rotiert. Ein Großkreis auf der xy-Ebene mit der z-Achse der Normalen entspricht dabei:

$$x = \cos(t) \quad y = \sin(t) \quad z = 0, \quad \text{wobei } 0 \leq t \leq 2\pi . \quad (4.11)$$

Jeder Punkt gegeben durch die Gleichung 4.11 muss nun rotiert werden, sodass die z-Achse der Normalen des passenden Großkreises entspricht. Da alle Punkte in einer Ebene liegen, kann für diesen Zweck die Reflektionsmatrix gegeben durch die Householdertransformation genutzt werden:

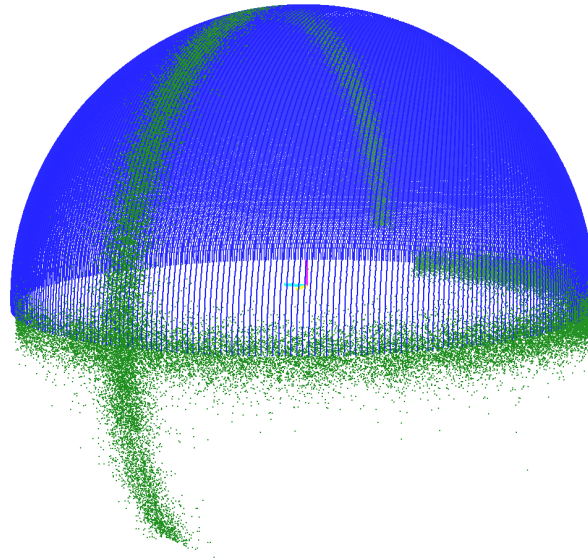
$$\mathbf{R} = \mathbf{I} - 2\mathbf{b}\mathbf{b}^T , \quad (4.12)$$

$$\text{mit } \mathbf{b} = \frac{(\mathbf{z} - \mathbf{n})}{\|\mathbf{z} - \mathbf{n}\|} , \quad (4.13)$$

wobei \mathbf{z} und \mathbf{n} definiert ist als:

$$\begin{aligned} \mathbf{z} &= (0 \ 0 \ 1)^T \\ \mathbf{n} &= (\cos \theta \sin \phi \quad \sin \theta \sin \phi \quad \cos \phi)^T . \end{aligned} \quad (4.14)$$

Mit der Rotationsmatrix \mathbf{R} können anschließend die Eingabepunkte gegeben durch die Normalen so gedreht werden, dass sie Großkreise bilden. Diese können mithilfe einer Ursprungesebenen-detektion, in diesem Fall einer RHT, berechnet werden, wobei die Zylinderachsen durch die Ebenennormalen dargestellt werden. [22]



created by 3DTK
<http://threadk.de>

Abbildung 4.2: Darstellung der sphärischen Zylindernormalen zweier simulierter Zylinder; blau: Darstellung gesampelte Einheitssphäre, grün: Darstellung sphärische Zylindernormalen

In Abbildung 4.2 sieht man die Normalen der Zylinder passend rotiert, sodass für jede Zylinderachse ein Großkreis auf der Gaußschen Sphäre entsteht. Die Besonderheit hierbei ist, dass durch die nicht ausgerichteten Normalen anstatt eines ganzen Großkreises ein halber Großkreis für eine Zylinderachse entsteht, wobei die Anzahl der Punkte des halben Großkreises, gleich der Anzahl der Punkte des Ganzen entspräche. Der Grund hierfür ist, dass die Hälfte der Zylindernormalen invertiert sind, da diese aufgrund der fehlenden Laserscannerposition nicht richtig ausgerichtet werden konnten. Dies stellt jedoch für die anschließende Hough-Transformation kein Problem dar, da trotzdem alle Normalen auf dem Großkreis liegen und somit für die passende Zylinderachse voten können.

Die für die Zylinderachsendetektion notwendige Vorbereitung der unsortierten Punktwolke wird im folgenden Pseudocode (1) zusammengefasst.

Pseudocode der Vorbereitung

Algorithm 1 Zylinderachsendetektion - Berechnung der sphärischen Normalen

Require: Eingabe: Punkteset \mathbf{S}

```

1: Berechne für jeden Punkt  $\mathbf{p}_i$  in  $\mathbf{S}$  seine Normale  $\mathbf{norm}_i$ 
2: Vektor  $\mathbf{z} = (0\ 0\ 1)$ 
3: Matrix  $\mathbf{b}[\mathbf{S.size()}, 3]$ 
4: for  $i = 0; i < \mathbf{S.size()}; i++$  do
5:    $[\theta_i, \phi_i] = \text{berechneKugelKoordinaten}(\mathbf{norm}_i)$ 
6:   Vektor  $\mathbf{n}_i[3]$ 
7:    $\mathbf{n}_i[0] = \cos(\theta_i) \sin(\phi_i)$ 
8:    $\mathbf{n}_i[1] = \sin(\theta_i) \sin(\phi_i)$ 
9:    $\mathbf{n}_i[2] = \cos(\phi_i)$ 
10:   $\mathbf{b}[i, :] = \mathbf{z} - \mathbf{n}_i$ 
11: end for
12:  $\mathbf{sphericalNormals}[\mathbf{S.size()}, 3]$ 
13: for  $i = 0; i < \mathbf{S.size()}; i++$  do
14:  Matrix  $\mathbf{R} = \text{IdentityMatrix}(3) - 2 \frac{\mathbf{b}[i,:]}{|\mathbf{b}|} \left( \frac{\mathbf{b}[i,:]}{|\mathbf{b}|} \right)^T$ 
15:   $\mathbf{sphericalNormals}[i, :] = \mathbf{R} * \mathbf{norm}_i$ 
16: end for
17: return  $\mathbf{sphericalNormals}$ 

```

4.1.2 Akkumulator

Bevor die eigentliche Hough-Transformation berechnet werden kann, muss zuerst der Akkumulator erstellt werden. Im Falle der Zylinderachsendetektion wird ein Akkumulator benötigt, der für die Normale eines Großkreises oder in diesem speziellen Implementationsfall, für die Normale einer Ursprungsebene geschnitten mit einer halben Einheitskugel voten kann. Im Falle von [22] wird der Hough-Raum durch eine Halbkugel dargestellt, die annähernd uniform gesampelt wird [17].

Für eine RHT wird dagegen meist ein sich mit der Laufzeit aufbauender Akkumulator verwendet. Da es jedoch für die Zylinderachsendetektion möglich ist den kompletten Hough-Raum mithilfe einer quantisierten halben Einheitskugel darzustellen, wird in diesem Fall darauf verzichtet. Aus diesem Grund wird der Akkumulator wie in [22] im Vorfeld generiert. Er besitzt ebenfalls die Struktur einer halben Einheitskugel, wird jedoch mit einem anderen Samplingansatz erstellt.

Im folgenden wird die Aufteilung des Akkumulators zuerst allgemein anhand einer Kugel erklärt. Die wichtigste Idee hinter dem Sampling ist, dass jede Akkumulatorzelle auf der Kugeloberfläche dieselbe Fläche besitzen soll. Hierfür wird die Sphäre mithilfe der Breiten- (θ) und Längengrade (φ) aufgeteilt. Das besondere dabei ist, dass die Auflösung der einzelnen Zellen je nach Position auf der Sphäre angepasst wird. Die Auflösung der Längengrade φ' ist dabei konstant und muss abhängig von dem Problem passend gewählt werden. Somit bestimmt φ' die Distanz zwischen

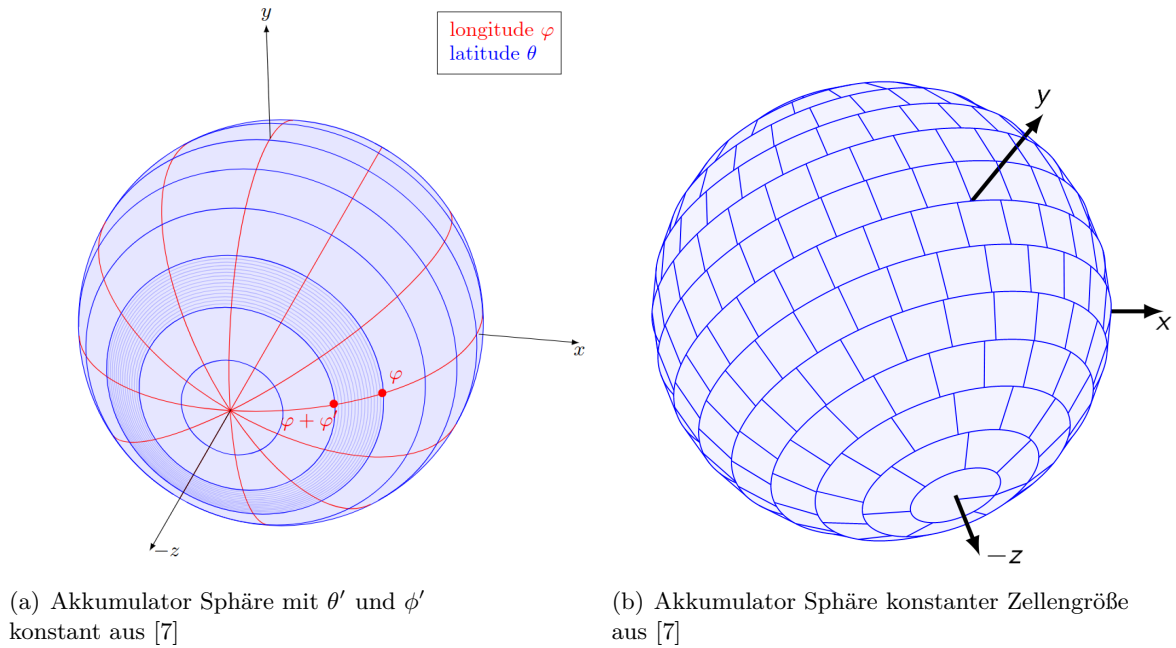


Abbildung 4.3: Sphärische Akkumulatoren

den einzelnen Längengraden, oder anders ausgedrückt, beschreibt ϕ' wie breit die einzelnen Kreisstücke sind. Passend zu dem jeweiligen Längengrad muss die Auflösung des Breitengrades bestimmt werden, sodass die Unterteilung der Kreisfläche zu gleich großen Zellen führt. Eine passende Diskretisierung für die Schrittweite θ' kann etwa über die Fläche des Längengrades bei φ_i bestimmt werden. Der größtmögliche Kreis befindet sich beim Äquator ($\varphi = 0$) und besitzt aufgrund der Einheitskugel eine maximale Länge von $l_{max_i} = 2\pi$. Die Länge des Kreises über φ_i ist gegeben durch $l_i = 2\pi(\varphi_i + \varphi')$. Die Schrittweite für θ kann somit berechnet werden mit:

$$\theta'_{\varphi_i} = \frac{2 \cdot \pi \cdot l_{max_i}}{l_i \cdot N_\theta} \quad , \quad (4.15)$$

wobei N_θ der Anzahl an Zellen entspricht. Mit einer derart angepassten Diskretisierung für die Schrittweite θ' ist es möglich, dass die Akkumulatorzellen alle eine nahezu identische Fläche besitzen. [7]

In Abbildung 4.3 wird die Akkumulatorkugel dargestellt, dabei ist links eine Sphäre zu erkennen deren Längen- und Breitengradauflösung gleich ist, wodurch die Flächen der unterteilten Zellen unterschiedlich groß sind. Hingegen wird rechts die Längengradauflösung passend zum Breitengrad verändert, sodass gleich große Zellen entstehen. [7]

Im Falle der Zylinderachsendetektion ist nur eine halbe Einheitskugel nötig, da ein beliebiger Vektor \mathbf{w} und $-\mathbf{w}$ derselben Zylinderachse entspricht. Aus diesem Grund wird analog zum oben beschriebenen Kugelsampling eine halbe Sphäre gesampelt und als Akkumulator genutzt. Dies verhindert, dass für jede Zylinderachse zwei Maxima im Akkumulator möglich sind. Nur am

Rande der Halbsphäre, also am Äquator der Kugel, besitzen Zylinderachsen zwei Maxima im Hough-Raum, wobei diese sich stets gegenüber liegen.

4.1.3 RHT Zylinderachsendetektion

Für die Zylinderachsendetektion wird in [22] eine Standard Hough-Transformation genutzt, um die Großkreise beziehungsweise die Schnittkreise einer Ursprungsebene mit der Einheitskugel zu detektieren. Da diese als Grundlage für die RHT dient, wird zuerst kurz die Standard Hough-Transformation zur Ursprungsebenendetektion beschrieben und anhand dieser die nötigen Veränderungen für die in dieser Arbeit verwendete RHT erklärt.

Ursprungsebenenbeschreibung

Eine Ebene im dreidimensionalen Raum wird oft beschrieben durch ihre gerichtete Distanz zum Koordinatenursprung ρ (Stützvektor), sowie durch ihre zwei Richtungsvektoren \mathbf{m}_x und \mathbf{m}_y :

$$\mathbf{z} = \mathbf{m}_x \cdot x + \mathbf{m}_y \cdot y + \rho \quad . \quad (4.16)$$

Ein Nachteil dieser Darstellung zeigt sich bei der Repräsentation von vertikalen Ebenen. In diesem Falle gehen die Richtungsvektoren gegen unendlich. Aus diesem Grund wird eine andere ebenfalls häufig genutzte Darstellung genutzt. Sei \mathbf{p} ein Punkt auf der Ebene, \mathbf{n} der Normalenvektor orthogonal zu Ebene und ρ der Abstand zum Koordinatenursprung, so ergibt sich:

$$\rho = \mathbf{p} \cdot \mathbf{n} = p_x n_x + p_y n_y + p_z n_z \quad . \quad (4.17)$$

Mithilfe der Winkel des Normalenvektors und des Koordinatensystems ist es somit möglich die Koordinaten von \mathbf{n} zu faktorisieren:

$$p_x \cdot \cos \theta \cdot \sin \phi + p_y \cdot \sin \theta \cdot \sin \phi + p_z \cdot \cos \phi = \rho \quad , \quad (4.18)$$

wobei θ den Winkel des Normalenvektors auf der xy-Ebene und ϕ den Winkel zwischen der xy-Ebene und dem Normalenvektor in z Richtung beschreibt. [7]

Da im Fall der Zylinderachsendetektion alle gesuchten Ebenen stets Ursprungsebenen sind, ist die Distanz zum Koordinatenursprung immer null, somit gilt $\rho = 0$. Der zweidimensionale Hough-Raum wird somit durch (ϕ, θ) definiert, wobei jeder diskretisierte Punkte des Hough-Raumes zu einer Ursprungsebene korrespondiert.

Standard Hough-Transformation

Für die in [22] vorgeschlagene Standard Hough-Transformation bedeutet dies, dass für jeden Punkt im rotierten sphärischen Normalenset eine Hough-Transformation berechnet wird. Gegeben sei als Beispiel ein beliebiger Punkt \mathbf{p} auf der Gaußschen Eingabesphäre. Bei einer Standard Hough-Transformation werden alle Akkumulatorzellen akkumuliert, die zu einer

Ursprungsebene korrespondieren auf der \mathbf{p} liegen kann. Dies führt zu einer 2D sinusoiden Kurve. Schneiden sich zwei Kurven von den beliebigen Punkten \mathbf{p}_i und \mathbf{p}_j , so beschreibt die dazu korrespondierende Schnittstelle im 2D-Hough-Raum die Ursprungsebene, die durch die zwei Punkte \mathbf{p}_i und \mathbf{p}_j aufgespannt wird.

Gegeben sei nun das Punkteset \mathbf{P} , welches auf der Gaußschen Eingabesphäre liegt. Alle Punkte $\mathbf{p}_i \in \mathbf{P}$ werden in den Hough-Raum transformiert. Je mehr Kurven sich in der Akkumulatorzelle $\mathbf{h}_j \in (\phi, \theta)$ schneiden, desto mehr Punkte liegen auf der Ursprungsebene repräsentiert durch \mathbf{h}_j . Mit jeder Kurve steigt die Wahrscheinlichkeit, dass die gefundene Ursprungsebene repräsentiert durch \mathbf{h}_j einem gesuchten Großkreis entspricht und damit eine Zylinderachse beschreibt. Nachdem alle Punkte gevotet haben, können die wahrscheinlichsten Zylinderachsen selektiert werden, dies wird genauer in Abschnitt 4.1.3 beschrieben. Im Pseudocode 2 wird die Standard Hough-Transformation zur Detektion von Zylinderachsen noch einmal kurz zusammengefasst. [7]

Auch wenn in [22] eine sequentielle zweiteilige Standard Hough-Transformation zur Zylinderdetektion genutzt wird, so ist diese zwar im Vergleich zu einer 5D Hough-Transformation deutlich weniger Raum- beziehungsweise Zeitkomplex, jedoch besitzt sie dennoch einige Nachteile. Zum einen ist die Anzahl der zu inkrementierenden Zellen des Akkumulators pro Eingabepunkt aufgrund der Standard Hough-Transformation sehr groß. Zum anderen kann es gut sein, dass es für die Zylinderachsendetektion nicht nötig ist alle Eingabepunkte zu betrachten. Aus diesem Grund wurde für diese Implementierung eine Randomized Hough-Transformation (RHT) verwendet. Diese besitzt einige Vorteile, die sich vor allem in der Laufzeit widerspiegeln, jedoch hat sie auch einen großen Nachteil im Vergleich zur Standard Hough-Transformation: Sie ist nicht deterministisch.

Algorithm 2 Zylinderachsendetektion - Standard Hough-Transformation

Require: Eingabe: Sphärische Normalen Set P

```

1: A = erstelleAkkumulator()
2: for jeden Punkt  $p_i \in P$  do
3:   for jede Zelle  $(\theta, \phi_i)$  in A do
4:     if  $p_i$  liegt auf Ursprungsebene  $(\theta, \phi_i)$  then
5:       A.inkrementZelle  $(\phi, \theta)$ 
6:     end if
7:   end for
8: end for
9: A.FilterFensterFunktion(FensterGröße)
10: A.sortiereVerbliebeneMaximaNachWahrscheinlichkeit()
11: return A.getMaxima()  $> t_{vmin}$ 

```

RHT

Die Randomized Hough-Transformation (RHT) besitzt den großen Vorteil, dass sie deutlich weniger Zellen des Hough-Raumes berühren muss. Um dies zu erreichen nutzt die RHT die Bedingung aus, dass eine Kurve mit n Parametern durch n Punkte definiert ist. Im Falle einer Zylinderachsendetektion beziehungsweise einer Ursprungsebene, bedeutet dies, dass immer zwei Eingabepunkte auf einen Punkt im Hough-Raum abgebildet werden. Dieser Punkt beschreibt die durch die zwei Punkte und dem Koordinatenursprung aufgespannte Ursprungsebene, wobei die Ebenennormale die Richtung der Zylinderachse beschreibt.

Im Gegensatz zu der Standard Hough-Transformation geht die RHT wie folgt vor: Zu Beginn jedes Schrittes werden zwei zufällige Punkte p_i und p_k ausgewählt. Da eine Ursprungsebene gesucht wird, ist zudem ein dritter Punkt p_0 definiert durch den Koordinatenursprung gegeben, wobei dieser als $(0\ 0\ 0)^T$ definiert ist. Aus diesen drei Punkten wird anschließend die Ebene berechnet, wobei gilt:

$$\begin{aligned} 0 &= \mathbf{n} \cdot \mathbf{p}_i \\ 0 &= ((\mathbf{p}_k - \mathbf{p}_0) \times (\mathbf{p}_i - \mathbf{p}_0)) \cdot \mathbf{p}_i \quad . \\ 0 &= (\mathbf{p}_k \times \mathbf{p}_i) \cdot \mathbf{p}_i \end{aligned} \quad (4.19)$$

Anschließend kann ϕ und θ der von \mathbf{p}_i und \mathbf{p}_k aufgespannten Ursprungsebene (beschrieben in 4.1.3) berechnet und die dazu korrespondierende Zelle im Akkumulator $\mathbf{A}(\phi, \theta)$ inkrementiert werden. Befindet sich ein Zylinder mit der Achsenrichtung (ϕ, θ) in der Szene, so wird nach einer bestimmten Anzahl an Iterationen ein Maximum bei der korrespondierenden Akkumulatorzelle (ϕ, θ) zu sehen sein, da viele Punkte sich auf der Mantelfläche des Zylinders befinden und somit die Wahrscheinlichkeit zwei sphärische Normalen zu selektieren, die für (ϕ, θ) stimmen, größer ist. So entsteht im Akkumulator bei der zur Zylinderachse korrespondierenden Zelle ein Maximum. [7]

Bei einer Ebenendetektion ist es zudem möglich eine Abstandsbedingung an die beiden zufällig ausgewählten Punkte zu stellen. Ist die Distanz zwischen zwei der drei Punkte zu groß, liegen sie häufig nicht auf einer Ebene. Aus diesem Grund kann man bei einer Ebenendetektion als Distanz Kriterium definieren: $dist_{max}(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3) \leq t_{dist}$ [7]

Dies ist im Falle der Zylinderachsendetektion nicht möglich, da alle rotierten sphärischen Normalenpunkte eines Zylinders auf einem Großkreis der Einheitssphäre liegen. Da ein Großkreis ein größtmöglicher Kreis auf einer Kugeloberfläche ist, ist die Distanz zwischen zwei auf dem Großkreis gegenüberliegenden Punkten maximal. Dies bedeutet man kann anhand der Distanz zwischen \mathbf{p}_i und \mathbf{p}_k nicht bestimmen, ob diese wahrscheinlich zu einer Zylinderachse gehören. Um Fehlern durch Sensorrauschen entgegenzuwirken, kann es jedoch sinnvoll sein ein minimales Distanz Kriterium zu definieren, so dass gilt $dist(\mathbf{p}_i, \mathbf{p}_k) > dist_{min}$.

In Abbildung 4.4 wird der Akkumulator nach einer erfolgreichen RHT dargestellt. Als Eingabe wurden die zwei Großkreise der zueinander senkrecht stehenden Zylinder verwendet (siehe Abb. 4.4). Die blauen Punkte beschreiben dabei Zellen, die nicht inkrementiert wurden, während rot Zellen mit vielen Akkumulationen anzeigt. Auffällig dabei ist, dass nicht nur zwei Maxima im Hough-Raum eingezeichnet sind, sondern drei. Wäre der Akkumulator nicht nur durch eine

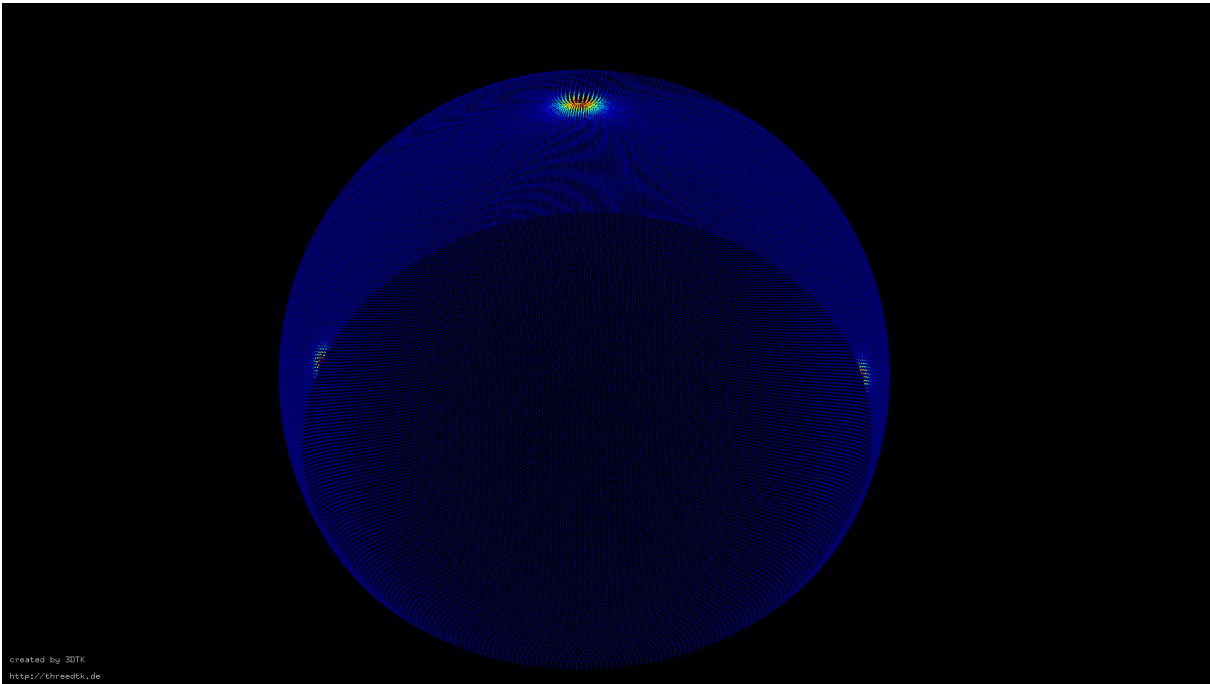


Abbildung 4.4: Darstellung des Akkumulators nach der Zylinderachsen RHT, wobei blau nicht inkrementiert und rot maximal viele Akkumulationen darstellen

Halbkugel, sondern durch eine ganze Einheitssphäre dargestellt, würden insgesamt vier Maxima detektiert werden. Dies liegt daran, dass eine der dargestellten Zylinderachsen am Rande der Halbsphäre liegt und somit auch ihre Invertierte im Akkumulator dargestellt wird. Diese möglichen doppelten Maxima im Hough-Raum dürfen für die nächsten Schritte nicht komplett vernachlässigt werden, da eine möglichst gute Zylinderachsenhypothese benötigt wird, um Zylinder robust zu detektieren.

Sliding Window Funktion

Nach der Hough-Transformation müssen aus dem Akkumulator die wahrscheinlichsten Zylinderachsen selektiert werden. In [22] wird hierfür ein einfacher Schwellwert verwendet. Alle Akkumulatorzellen deren Voteanzahl größer als der Schwellwert ist, werden als Hypothese für die Zylinderachsenrichtung betrachtet. Dieses Verfahren ist relativ einfach und schnell, kann jedoch zu einigen Problemen führen. Sind in einem Laserscan etwa sehr viele Zylinder mit derselben Achse \mathbf{n}_i und nur ein Zylinder mit einer anderen Achsenrichtung \mathbf{n}_k vorhanden, wobei die Anzahl der gesampelten Punkte auf jedem Zylinder gleich sei, somit voten deutlich mehr sphärische Normalenpunkte für \mathbf{n}_i als für \mathbf{n}_k . Aufgrund der Diskretisierung des Hough-Raumes, dem Rauschen auf den Eingabepunkten und Ungenauigkeiten bei der Bestimmung der Normalen, gerade an den Zylinderkanten, voten die Punkte einer beliebigen Zylinderachse jedoch nicht nur für eine Akkumulatorzelle, sondern für mehrere. Um beide Zylinderachsen \mathbf{n}_k und \mathbf{n}_i

mit nur einem Schwellwert zu selektieren, muss der Schwellwert so klein sein, dass mindestens eine zu \mathbf{n}_k korrespondierende Zelle einen größeren Score aufweist als der Schwellwert. Dies kann jedoch wiederum dazu führen, dass der Schwellwert für die Achse \mathbf{n}_i zu klein ist, so dass nicht nur eine Akkumulatorzelle für \mathbf{n}_i selektiert wird, sondern das alle Zellen im nahen Umkreis um das eigentlich Maxima ebenfalls größer als der Schwellwert sind.

Um dieses Problem zu vermeiden ist es ratsam nicht nur nach den Zellen mit den maximalen Score zu suchen, sondern immer das Maximum einer schmalen Region des Akkumulators zu betrachten. Eine Möglichkeit hierfür ist etwa die „Sliding Window“ Prozedur zur Peak-Detektion [14]. Hierbei wird ein kleines zweidimensionales Fenster definiert, das den kompletten Peak Spread überlagern kann. Die hervorstechendste Ebene korrespondiert dabei mit dem Mittelpunkt des Quadrats im Hough-Raum mit dem maximalen Peak. Die einzelnen Schritte der RHT werden als Pseudocode in 3 noch einmal kurz dargestellt.

Pseudocode der RHT-Zylinderachsendetektion

Algorithm 3 Zylinderachsendetektion - RHT

Require: Eingabe: Sphärische Normalen Set \mathbf{P}

```

1:  $\mathbf{A}$  = erstelleAkkumulator()
2: while  $t < t_{max}$  do
3:   Wähle  $\mathbf{p}_1, \mathbf{p}_2 \in \mathbf{P}$  zufällig
4:    $dist$  = berechneDistanzZwischen2Punkten( $\mathbf{p}_1, \mathbf{p}_2$ )
5:   if  $dist < \text{Schwellwert}$  then
6:     continue
7:   end if
8:    $[\phi, \theta]$  = berechneUrsprungsEbene( $\mathbf{p}_1, \mathbf{p}_2$ )
9:   if not liegtAufAkkumulatorHalbkugel( $\phi$ ) then
10:     $[\phi, \theta]$  = invertierteEbenenNormale( $\phi, \theta$ )
11:   end if
12:    $\mathbf{A}$ .inkrement( $\phi, \theta$ )
13:    $t++$ 
14: end while
15: Zylinderachsen  $\mathbf{z}$  = SlidingWindowFilter( $\mathbf{A}$ , fensterGröße)
16:  $\mathbf{z}$  = sortiereAchsenNachWahrscheinlichkeit( $\mathbf{z}$ )
17: return  $\mathbf{z}$ 

```

4.1.4 Generierung eines orthonormalen Koordinatensystems

Nachdem die Auswertung des Hough-Raumes erfolgt ist, werden die selektierten Zylinderachsen einzeln der Kreisdetektion übergeben. Vor der Übergabe muss für die entsprechende Achse ein orthonormales Koordinatensystem bestimmt werden, sodass die Kreisdetektion möglichst schnell die 3D-Punkte entlang der Zylinderachse auf eine Ebene projizieren kann. Dabei gilt für die dreidimensionalen orthonormierten Basisvektoren \mathbf{u} , \mathbf{v} und \mathbf{w} :

$$\|\mathbf{u}\| = \|\mathbf{v}\| = \|\mathbf{w}\| = 1 \quad (4.20)$$

und

$$\mathbf{u} \cdot \mathbf{v} = \mathbf{v} \cdot \mathbf{w} = \mathbf{w} \cdot \mathbf{u} = 0 \quad . \quad (4.21)$$

Für eine rechtshändige Basis gilt:

$$\mathbf{w} = \mathbf{u} \times \mathbf{v} \quad . \quad [18] \quad (4.22)$$

Da das orthonormale Koordinatensystem in diesem Fall für eine Projektion der Punktwolke entlang der Zylinderachse genutzt werden soll, ist es essentiell dass die Richtung der Zylinderachse einem orthonormierten Basisvektor entspricht. Für die Berechnung dieses Koordinatensystems werden in [22] zwei mögliche Implementierungen vorgeschlagen. Zum einen wird die Gram-Schmidt Orthogonalization, zum anderen eine Lösung mithilfe einer SVD (singular value decomposition) vorgeschlagen.

Für beide Verfahren müssen zuerst N nächste Nachbarpunkte für die Zylinderachse gesucht werden, wobei [22] hierfür einen approximierten uniformen Sampling Ansatz nutzt. Aus diesen N Vektoren, soll eine orthonormierte Basis konstruiert werden. Der klassische Gram-Schmidt Ansatz startet dabei mit einem Vektor und expandiert den Unterraum in jedem Schritt um eine Dimension. Numerisch betrachtet ist dieser Ansatz jedoch schlecht, denn die Rundungsfehler addieren sich auf, weswegen zumindest ein modifizierter Gram-Schmidt Algorithmus verwendet werden sollte (beschrieben in [9]). Für die zweite Lösung hingegen muss mithilfe der N 3-dimensionalen Vektoren eine $M \times N$ Matrix A gebildet und diese mithilfe einer SVD gelöst werden [21].

Beide Verfahren haben den großen Nachteil, dass sie vergleichsweise rechenintensiv für das relativ einfache Problem der Berechnung eines orthonormalen Koordinatensystems für eine Zylinderachse sind. Insbesondere wenn man bedenkt, dass man mit der RHT-Zylinderachsendetektion nur eine starke Hypothese, nicht jedoch die exakten Zylinderachsen berechnet. Dies kann dazu führen, dass neben den richtigen Zylinderachsen auch Falsche oder Doppelte, aufgrund von Rauschen, Ausreißern und Berechnungsungenauigkeiten bei der sphärischen Normalenkalkulation, gefunden werden. Die SVD muss also im Optimalfall mindestens einmal für alle Zylinderachsenrichtungen, in der Realität jedoch deutlich häufiger berechnet werden.

Aus diesem Grund wurde für die hier vorgestellte Implementierung, ein anderes Vorgehen verwendet. Dieses bezieht nicht die Nachbarn eines Maximums mit ein, sondern berechnet das Koordinatensystem direkt aus der Achse, weswegen es sinnvoll ist die Auflösung des

Akkumulators feiner zu wählen und die Maxima nicht mit einem einfachen Schwellwert, sondern mit einem sliding Window Filter zu detektieren. Eine feinere Auflösung des Akkumulators ist aufgrund der RHT möglich, denn die zu akkumulierenden Zellen für zwei randomisiert gewählte Punkte ändern sich, was im Gegensatz zur Standard Hough-Transformation nicht der Fall ist.

Somit ist es möglich, das orthonormale Koordinatensystem direkt aus der Zylinderachsenrichtung \mathbf{n} abzuleiten. Da die Zylinderachsenrichtung aufgrund der Einheitskugel des Hough-Raumes die Länge 1 hat, gilt:

$$\mathbf{w} = \frac{\mathbf{n}}{\|\mathbf{n}\|} = \mathbf{n} \quad . \quad (4.23)$$

Anschließend wird ein beliebiger Vektor \mathbf{t} definiert, der nicht kollinear zu \mathbf{w} ist. Mithilfe dieser Vektoren und dem Kreuzprodukt ist es möglich den Einheitsvektor \mathbf{u} senkrecht zu \mathbf{w} zu bestimmen:

$$\mathbf{u} = \frac{\mathbf{t} \times \mathbf{w}}{\|\mathbf{t} \times \mathbf{w}\|} \quad . \quad (4.24)$$

Wäre \mathbf{t} kollinear zu \mathbf{w} , so würde der Nenner 0 sein und die Gleichung nicht definiert. Aus diesem Grunde sollte \mathbf{t} auch nicht fast kollinear zu \mathbf{w} sein, da sonst das Ergebnis eine schlechte Genauigkeit besitzt. Anschließend kann man aus \mathbf{w} und \mathbf{u} mithilfe des Kreuzproduktes \mathbf{v} erzeugen:

$$\mathbf{v} = \mathbf{w} \times \mathbf{u} \quad . \quad (4.25)$$

Dieses Vorgehen ermöglicht es sehr einfach und ohne großen Rechenaufwand mithilfe der Zylinderachse ein orthonormales Koordinatensystem aufzubauen. [18]

4.2 Positions- und Radiusberechnung

In Abschnitt 4.1 wurde beschrieben wie man eine starke Hypothese für die Richtung einer beliebigen Zylinderachse findet. Somit wurden mit diesem Schritt zwei der fünf gesuchten Zylinderparameter bestimmt. Die fehlenden drei Parameter beschreiben die Position und den Radius des Zylinders. Sie müssen mithilfe der zweiten Hough-Transformation für jede gefundene Zylinderachse neu bestimmt und berechnet werden. Aus diesem Grund muss die Hough-Transformation zur Bestimmung von Radius und Zylinderposition wesentlich häufiger berechnet werden, als die für die Zylinderachsendetektion.

Dabei geht die zweite Hough-Transformation für eine Zylinderachsenrichtung \mathbf{n} folgendermaßen vor: Zuerst werden alle möglichen Zylinderpunkte entlang \mathbf{n} auf eine Ebene projiziert. So entsteht aus einem Zylinder ein Kreis. Dieser kann anschließend mithilfe einer Hough-Transformation gefunden werden. Der Kreisradius entspricht dabei dem Zylinderradius. Wird der Kreismittelpunkt zurück in den dreidimensionalen Raum transformiert, so kann direkt die Position des Zylinders daraus erhalten werden.

Im Folgenden werden die einzelnen Implementierungsschritte der Zylinderposition sowie der Radiusberechnung genauer beschrieben.

4.2.1 Vorbereitung - Selektion wahrscheinlicher Zylinderpunkte

Zu Beginn der Positions- und Radiusbestimmung ist es wichtig die Punkte des Laserscans vorzusortieren. Nur Punkte deren sphärische Normalen auf dem Großkreis senkrecht zur Zylinderachsenrichtung liegen, dürfen der zweiten Hough-Transformation übergeben werden. Aufgrund von Rauschen und kleineren Fehlern in der Normalenberechnung liegen jedoch nur die wenigsten sphärischen Normalenpunkte exakt auf der Großkreisebene.

Aus diesem Grund ist es sinnvoll einen Schwellwert zu definieren, der die maximale Distanz zur Großkreisebene begrenzt. Der Schwellwert sollte dabei groß genug sein, um die meisten fehlerbehafteten sphärischen Normalenpunkte, die für die Ebene voten, zu selektieren, gleichzeitig sollte er klein genug gewählt sein, um zwischen den zwei ähnlichsten Großkreisen unterscheiden zu können. Da sowohl das Messrauschen, als auch der minimale Winkelunterschied zwischen zwei zu findenden Zylinderachsen, von Scan zu Scan unterschiedlich sind, muss dieser Parameter je nach Datensatz passend gewählt werden. Für die Experimente und Auswertungen in dieser Arbeit wurde der Schwellwert auf einen maximalen Abstand von 0,01 gesetzt.

Die Distanz d eines sphärischen Normale \mathbf{p}_n zu einem Großkreis beziehungsweise zu einer Ursprungsebene senkrecht zur Zylinderachse \mathbf{n} wird wie folgt berechnet:

$$d = \frac{|\mathbf{n} \cdot \mathbf{p}_n|}{|\mathbf{n}|} = |\mathbf{n} \cdot \mathbf{p}_n| \quad [20]. \quad (4.26)$$

Da die Zylinderachse mithilfe der Einheitssphäre berechnet wurde, ist sie bereits normalisiert, sodass man bei der Distanzberechnung nicht durch ihren Betrag dividieren muss. Ist d kleiner als der Schwellwert, so wird der 3D-Punkt \mathbf{p} mit der sphärischen Normale \mathbf{p}_n für die Kreisdetektion genutzt, ist er größer wird er für diese spezielle Zylinderachse verworfen.

4.2.2 Vorbereitung - Projektion der Zylinderpunkte

Alle dreidimensionalen Punkte, deren sphärische Normalen für die Zylinderachse im ersten Schritt voten, werden im Folgenden zusammengefasst in dem Set \mathbf{P}_c . Jeder Punkt $\mathbf{p}_i \in \mathbf{P}_c$ muss vor der Hough-Transformation entlang der Zylinderachse auf die Ebene projiziert werden. Für eine möglichst effiziente Projektion wird ein orthonormales Koordinatensystem verwendet, dessen Aufbau in Abschnitt 4.1.4 beschrieben wurde. Es wird direkt nach der Zylinderachsen-detektion berechnet und an die Kreisdetektion übergeben.

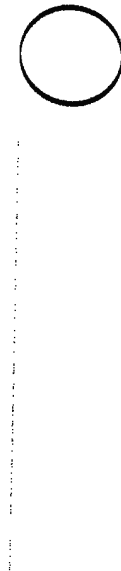
Sei \mathbf{u} , \mathbf{v} , \mathbf{w} die orthonormierte Basis des Zylinderachsenkoordinatensystems, wobei \mathbf{w} gleich der Zylinderachsenrichtung \mathbf{n} ist. Sei außerdem \mathbf{p} ein beliebiger Punkt aus \mathbf{P}_c . Dann kann der korrespondierende projizierte Punkt \mathbf{p}_{proj} folgendermaßen berechnet werden:

$$\mathbf{p}_{\text{proj}} = (\mathbf{p} \cdot \mathbf{u})\mathbf{u} + (\mathbf{p} \cdot \mathbf{v})\mathbf{v} \quad . \quad (4.27)$$

Für die anschließende Hough-Transformation ist es sinnvoll die noch dreidimensionalen kartesischen Koordinaten $\mathbf{p} = (p_x, p_y, p_z)$ in zweidimensionale Ebenenkoordinaten $\mathbf{p}_{2D} = (p_u, p_v)$ umzurechnen:

$$\begin{aligned} p_u &= \mathbf{P}_{\text{proj}} \cdot \mathbf{u} \\ p_v &= \mathbf{P}_{\text{proj}} \cdot \mathbf{v} \end{aligned} \quad (4.28)$$

In Abbildung 4.5 ist die Ebenenprojektion eines der Beispielzylinder aus Abschnitt 4.1 zu sehen. Der aus den projizierten Mantelpunkten entstandene Kreis ist deutlich zu erkennen. Gleichzeitig fällt auf, dass einige Punkte des zweiten Zylinders ebenfalls selektiert sind. Diese sind zwar nur in geringer Zahl vertreten, sollten jedoch bei der Implementierung der Kreisdetektion beachtet werden.



created by 3DTK
<http://threedtk.de>

Abbildung 4.5: Projektion eines Zylinders entlang seiner Achse und Ausreißer durch zweiten Zylinder senkrecht zu erstem

4.2.3 Standard Hough-Transformation

Wie eingangs schon erwähnt verwendet [22] eine Standard Hough-Transformation zur Detektion der projizierten Zylindern, beziehungsweise Kreise. Hierfür wird ein Ansatz nach [13] implementiert. Im Folgenden wird dieses Verfahren kurz erklärt, um anschließend seine Schwächen aufzuzeigen und diese zu verbessern.

Akkumulator

Da es sich bei der Kreisdetektion um eine dreidimensionale Hough-Transformation handelt, gilt es die drei Parameter (Radius und Kreismittelpunkt (c_u, c_v)) im Akkumulator darzustellen und zu diskretisieren. Hierfür muss der Nutzer vor dem eigentlichen Beginn der Standard Hough-Transformation den Radiusbereich der Zylinder $[r_{min}, r_{max}]$, sowie die Auflösung der Radius-schritte r' festlegen. Es soll dabei stets gelten $r_{i+1} = r_i + r'$.

Anschließend muss der Wertebereich von \mathbf{u} und \mathbf{v} des Akkumulators für jedes r_i so festgelegt werden, dass für jeden möglichen Kreis mit Mittelpunkt (c_u, c_v) in der Akkumulatorebene korrespondierend zum Radius r_i gevotet werden kann. Die Grenzen und Auflösung von \mathbf{u} und \mathbf{v} können entweder ebenfalls fest über den Nutzer definiert werden oder müssen mithilfe der projizierten Punkte und r_i berechnet werden. Ersteres hat den Nachteil, dass der Nutzer einen festen Akkumulatorraum für alle Achsen und Radii im Vorfeld definieren muss. Dies kann dazu führen, dass der Akkumulator zu groß oder im schlimmsten Fall zu klein ist, um alle möglichen Kreise zu finden.

Eine Möglichkeit die Größe des Hough-Raumes dynamisch festzulegen, kann etwa mithilfe der projizierten Punkte erfolgen. Für beide Richtungen u und v werden dafür die minimalen beziehungsweise maximalen Werte der projizierten Punkte vermerkt, sodass die Wertebereiche $[u_{min}, u_{max}]$ und $[v_{min}, v_{max}]$ gegeben sind. Die Ecken einer rechteckigen Akkumulatorebene für r_i , wobei alle Hough-Transformationen der projizierten Punkte dargestellt werden sollen, können beschrieben werden durch:

$$\begin{bmatrix} [u_{max} + r_i, v_{min} - r_i] & [u_{max} + r_i, v_{max} + r_i] \\ [u_{min} - r_i, v_{min} - r_i] & [u_{min} - r_i, v_{max} + r_i] \end{bmatrix} . \quad (4.29)$$

Diese Eingrenzung hat den Nachteil, dass die einzelnen Ebenen des Akkumulators unterschiedlich groß sind, wodurch entweder die Auflösung oder die Anzahl an Samplingpunkten der einzelnen Ebenen unterschiedlich groß sein muss. Dies ist für die Implementierung nicht optimal und lässt sich zu dem Vermeiden sollte folgende Bedingung erfüllt sein:

$$\begin{aligned} u_{max} &> c_{u_{max}} \\ u_{min} &< c_{u_{min}} \\ v_{max} &> c_{v_{max}} \\ v_{min} &< c_{v_{min}} \end{aligned} . \quad (4.30)$$

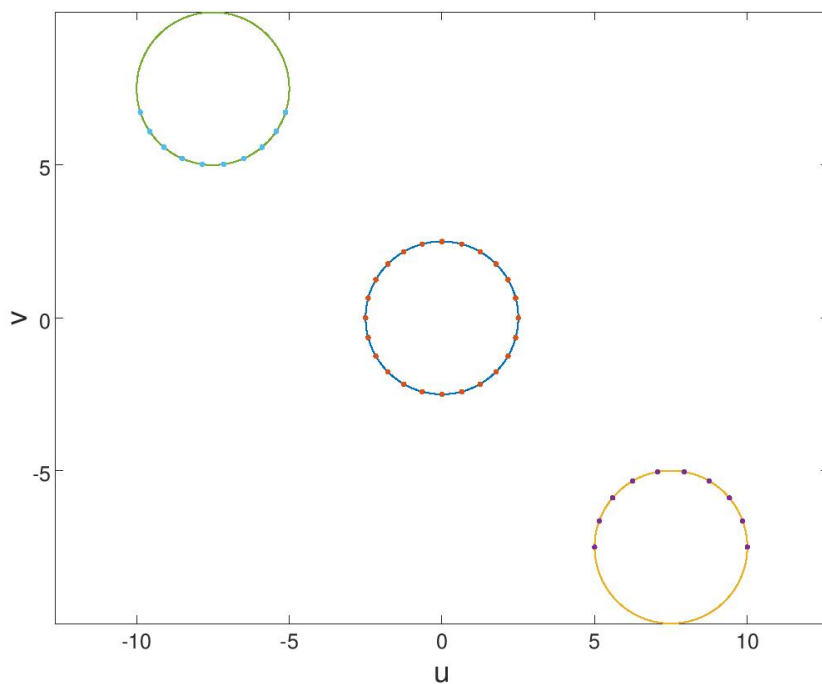
Wobei $[c_{u_{min}}, c_{u_{max}}]$ und $[c_{v_{min}}, c_{v_{max}}]$, den maximalen beziehungsweise minimalen Wert in \mathbf{u} oder \mathbf{v} aller Kreismittelpunkte beschreiben. Ist dieses Bedingung erfüllt, so können die Eckpunkte einer Akkumulatorebene definiert werden als:

$$\begin{bmatrix} [u_{max}, v_{min}] & [u_{max}, v_{max}] \\ [u_{min}, v_{min}] & [u_{min}, v_{max}] \end{bmatrix}. \quad (4.31)$$

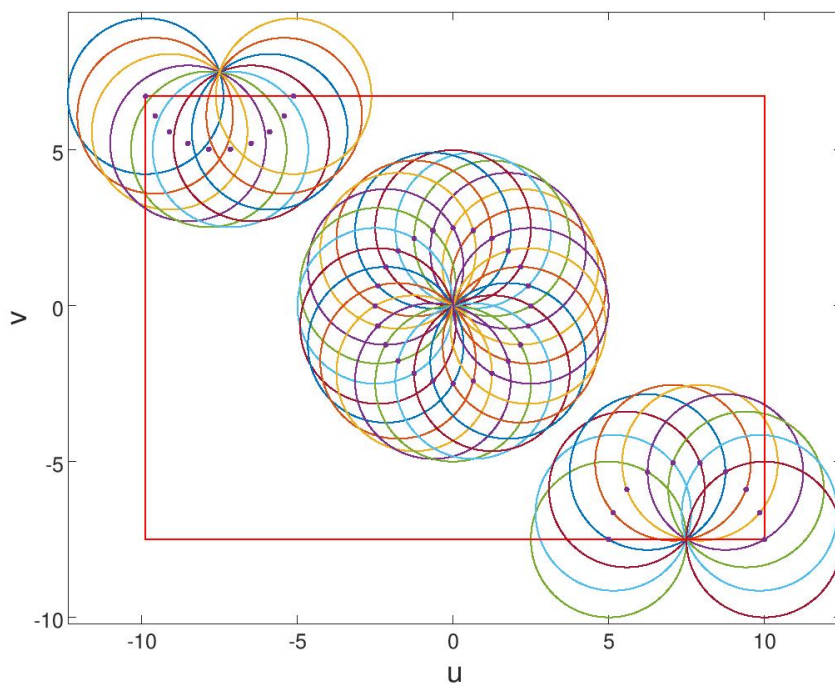
Ein derart aufgespannter Akkumulator kann nicht mehr die transformierten Randpunkte komplett darstellen, jedoch liegen alle Kreismittelpunkte im vom Akkumulator quantisierten Hough-Raum Bereich, solange die Bedingung nicht verletzt ist.

In Abbildung 4.6 sind im oberen Bild drei Beispielkreise eingezeichnet. Während der Kreis in der Mitte komplett gesampelt ist und somit dem in [22] vorgestellten Abbild entspricht, ist für den gelben Randkreis nur die Hälfte und für den Grünen Randkreis nicht einmal die Hälfte gesampelt. Mit rot ist der Hough-Raum eingezeichnet, der möglich ist sollte die Bedingung 4.30 erfüllt sein. Dies ist jedoch nicht der Fall, da gilt $v_{max} < c_{v_{max}}$ und $v_{min} = c_{v_{min}}$. Ersteres führt dazu, dass der Kreismittelpunkt des grünen Kreises sich außerhalb des Hough-Raumes befindet. Letzteres führt zu einer Randbedingung, weswegen der Mittelpunkt des gelben Kreises auf dem Rand des Akkumulators liegt. Bei einer Zylinderdetektion kann aufgrund der fehlerbehafteten Zylinderachse diese Randbedingung schnell dazu führen, dass der Kreismittelpunkt außerhalb des Akkumulators liegt, daher wird sie aus der Bedingung ausgeschlossen.

Ein großer Nachteil bei diesem Verfahren ist, dass der Aufbau des Akkumulators auch mit dieser Vereinfachung noch nicht optimal ist. Dies liegt daran, dass nicht alle Punkte aus P_c Zylinderpunkten mit passender Achse entsprechen. Oft werden auch einige Punkte ausgewählt, deren sphärische Normale zufällig für die Zylinderachsenrichtung gestimmt haben. Diese falsch selektierten Punkte aus P_c können, wenn sie weit von den Kreisen entfernt sind, den Akkumulatorraum unnötig aufspannen und die Auflösung des Raumes durch das somit begrenzte Sampling deutlich verschlechtern.



(a) Gesampelte Kreise Beispiel



(b) Hough-Transformationen der Kreispunkte und Hough-Raum (rot) berechnet ohne Radius

Abbildung 4.6: Darstellung gesampelter Beispielkreise und ihrer Hough-Transformationen

ENTWICKLUNG EINER RANDOMISIERTEN HOUGH-TRANSFORMATION ZUR
ZYLINDERDETEKTION IN LASERSCANS

Voting

Im Falle der Standard Hough-Transformation zur Kreisdetektion, muss analog zur Ebenendetektion für jeden Punkt \mathbf{P}_{2D} , berechnet aus dem Set P_c , eine Hough-Transformation bestimmt werden. Dabei entspricht die Hough-Transformation eines beliebigen Punktes $\mathbf{p}_k = (p_u, p_v)$ auf der Akkumulatorebene korrespondierend zum Radius r_i einem Kreis mit dem Kreismittelpunkt (p_u, p_v) und dem Radius r_i . Es wird also für alle Zellen gevotet, die durch die Funktion:

$$(r_i \cos \omega + p_u, r_i \sin \omega + p_v) \quad (4.32)$$

berührt werden. Dieses Voting wird für alle Ebenen so wie projizierten Punkte wiederholt, bis jeder Punkt auf jeder Akkumulatorebene für einen Kreis gevotet hat.

Auswertung Hough-Raum

Nach dem Voting wird in [22] der maximale Peak im Hough-Raum gesucht. Dieser Kreis korrespondiert zu der Position sowie dem Radius des wahrscheinlichsten Zylinders. Anschließend werden die Zylinderpunkte gelöscht und falls noch weitere Punkte vorhanden sind, wird eine neue Orientierung betrachtet [22]. Mögliche weitere Peaks, die von Zylindern mit derselben Achse stammen, werden nach dem beschriebenen Kreisdetektionsalgorithmus (Seite 63 in [22]) nicht ausgewertet. Da in den realen Beispielen jedoch auch Zylinder mit scheinbar der gleicher oder zumindest sehr ähnlicher Achse gefunden und selektiert werden, scheinen mehrere Hypothesen für eine Zylinderachsenrichtung in [22] zulässig und gewünscht zu sein. Dies wäre theoretisch aufgrund des genutzten lokalen Schwellwerts zur Auswertung des Akkumulators möglich. Jedoch würde es auch bedeuten, dass für jeden Zylinder in der Szene eine eigene Standard-Hough-Transformation berechnet werden muss.

4.2.4 Randomized Hough-Transformation

Um die meisten der in Abschnitt 4.2.3 beschriebenen Probleme zu lösen oder zumindest zu minimieren, wird in dieser Arbeit auf eine Standard Hough-Transformation für die Kreisdetektion verzichtet. Stattdessen wird ebenso wie für die Zylinderachsendetektion eine RHT implementiert. Dies hat einige große Vorteile: Die RHT kann deutlich schneller als eine Hough-Transformation sein, was wichtig ist, da die Kreisdetektion sehr häufig berechnet werden muss. Des Weiteren muss der Akkumulator nicht vor der Hough-Transformation bestimmt werden, sondern kann dynamisch während der RHT aufgebaut werden.

Ein weiterer Unterschied in der Implementierung zwischen dem Ansatz aus [22] und dieser Variante zeigt sich in der Auswertung des Hough-Raumes. Anstatt nur das beste Maximum des Akkumulators zu berücksichtigen, werden alle vielversprechenden Kreise während der RHT direkt validiert und im positiven Fall mithilfe eines Least-Square Ansatzes möglichst gut in die projizierten Punkte der nächsten Umgebung gelegt. Anschließend werden nur die projizierten Punkte gelöscht und mit der Kreisdetektion für die entsprechende Achse fortgesetzt, die ursprünglichen dreidimensionalen Punkte werden nicht gelöscht.

Es ist nicht möglich die Punkte aus der ursprünglichen Punktwolke zu löschen, da es für zwei beliebige Zylinder \mathbf{z}_i und \mathbf{z}_k mehrere Detektionen geben kann, falls der Winkel zwischen den Zylinderachsen \mathbf{w}_i und \mathbf{w}_k klein ist. So kann es möglich sein \mathbf{z}_i sowohl bei einer Projektion entlang \mathbf{w}_i als auch \mathbf{w}_k als Kreis zu detektieren, wobei letzteres zu einem größeren Fehler in den Zylinderparametern führt. Somit ist es nicht möglich nach einer erfolgreichen Detektion die Punkte aus der Punktwolke zu entfernen, da ein weiteres besseres Matching mit einer anderen Achse möglich sein kann. Aus diesem Grund müssen am Ende des Algorithmus alle Mehrfachdetektionen identifiziert und der Konflikt durch das Löschen der schlechteren Zylinder gelöst werden.

Im folgenden Abschnitt wird der sich dynamisch aufbauende Akkumulator, sowie das Voting der RHT und schließlich die Validierung der vielversprechenden Kreise beschrieben. Der bereits erwähnte Least-Square Ansatz zum besseren Fitten der Kreise wird erst im darauf folgenden Abschnitt 4.2.5 und die Schnittpunktdetektion erst im nächsten Kapitel 4.3.2 beschrieben.

Akkumulator und Voting

Im Gegensatz zum Akkumulator der Standard Hough-Transformation muss der Akkumulator für die RHT nicht zu Beginn des Votings erstellt sein. Stattdessen wird der Akkumulator gemeinsam mit der fortschreitenden RHT aufgebaut. Das einfache Grundprinzip funktioniert dabei wie folgt: In jeder RHT Iteration werden drei zufällige Punkte aus dem Set \mathbf{P}_{2D} ausgewählt. Sind diese nicht kollinear wird aus ihnen ein Kreis \mathbf{K} mit dem Kreismittelpunkt $\mathbf{c}_m = (u_m, v_m)$ und dem Radius r berechnet, die drei Parameter werden anschließend auf l Nachkommastellen gerundet. Die gerundeten Kreisparameter (u'_m, v'_m, r') beschreiben eine Zelle im Akkumulator. Wurde noch kein Kreis mit den Parametern (u'_m, v'_m, r') gefunden, so wird ein neuer Eintrag mit einem Zählerwert gleich 1 im Akkumulator erstellt. Ist bereits ein Eintrag mit den passenden Parametern vorhanden, so wird dessen Zähler inkrementiert. [10]

Mit dieser Strategie ist es möglich nur die notwendigen Zellen im Akkumulator während des Votings zu erstellen und zu inkrementieren. Es ist nicht nötig bereits zu Beginn den kompletten Akkumulator aufzuspannen. Dies bringt den großen Vorteil, dass die Größe des Akkumulators erweiterbar ist und es somit nicht zu einer Fehl- oder Falschdetektion aufgrund eines zu klein gewählten Akkumulators kommt. Neben der Größe kann zudem die Genauigkeit, mit welcher die Kreise detektiert werden, angepasst werden. Dies ist für den vorgestellten Algorithmus allerdings nicht von großer Bedeutung, da eine sehr grobe, dafür jedoch auch schnellere RHT verwendet wird. Ein solches Vorgehen ist möglich da anschließend validierte Kreise mithilfe eines Least-Square Ansatzes auf ihre nächste Umgebung feiner bestimmt werden.

Validierung

Nicht nur das Erstellen des Akkumulators erfolgt zur Laufzeit, sondern auch die Validierung vielversprechender Kreise wird bereits während der RHT berechnet. Sobald ein Eintrag einen Zähler gleich des Validierungsschwellwert \mathbf{v} besitzt, wird der dazu passende Kreis validiert. [10]

Hierfür werden alle Punkte aus \mathbf{P}_{2D} gewählt, deren Distanz zu dem Kreis, beschrieben durch die Parameter (u'_m, v'_m, r') , kleiner ist als ein Schwellwert t_{dis} . Dabei sollte t_{dis} größer als die durch die Rundung entstehende Ungenauigkeit sein. Zudem wird die Anzahl aller Punkte \mathbf{P}_{inlier} die sich innerhalb des Kreises befinden, jedoch eine größere Distanz als t_{dis} aufweisen, gezählt.

Sobald \mathbf{P}_{circ} erstellt und die Anzahl von \mathbf{P}_{inlier} bestimmt ist, kann die Validierung beginnen. Diese besteht aus zwei Schritten, einer schnellen, groben Vorsortierung und einer feineren jedoch deutlich rechenintensiveren Auswertung. Für Erstere wird zuerst überprüft, ob die Anzahl von \mathbf{P}_{circ} größer ist als \mathbf{P}_{inlier} , um sicherzustellen, dass kein zufälliger Kreis in einer Ansammlung von zufälligen Punkten detektiert wird. Eine mögliche Erweiterung dieser Abfrage kann helfen, einige sehr unwahrscheinliche Kreise bereits im Vorfeld auszusortieren, so dass für diese keine feinere Auswertung berechnet werden muss. Hierfür wurden zwei weitere Abfragen eingebaut, welche beide mit einem vom Nutzer vorgegebenen Schwellwert arbeiten. Zum einen wird überprüft, ob die Anzahl an Punkten \mathbf{P}_{circ} größer als t_{min} ist. Dieser Schwellwert testet ob eine Mindestanzahl an Punkten auf dem detektierten Kreis liegt. Für die zweite Abfrage werden die Kosten des Kreises, dargestellt durch die Anzahl der Punkte \mathbf{P}_{circ} dividiert durch r' , berechnet. Je kleiner die Kosten sind, desto größer ist die Wahrscheinlichkeit einen Kreis auf eine zufällige Punkteverteilung gefittet zu haben. Aus diesem Grund ist es sinnvoll zumindest einen kleinen Schwellwert t_{cost} für die minimalen Kosten zu setzen. Für die Auswertung in Kapitel 5 wurde $t_{min} = 250$ und $t_{cost} = 5$ gewählt. Eine grobe Vorsortierung wie diese, nimmt aufgrund des sehr klein gewählten Schwellwert keinen Einfluss auf das Ergebnis der Kreisdetektion, da derart unwahrscheinliche Kreise durch den zweiten Validierungsschritt ebenfalls aussortiert werden. Sie ist jedoch deutlich schneller und kann so helfen die Kreisdetektion etwas zu beschleunigen.

Für den zweiten Validierungsschritt wird der Kreis in n_{seg} gleichgroße Kreissektoren aufgeteilt. Dabei soll die Länge des Kreisbogens b stets vom Benutzer definiert sein, während der Winkel $\Delta\theta$ des Kreissektors in rad sich in Abhängigkeit zu r' verändert. Da die Länge des Kreisbogens definiert ist als:

$$b = 2\pi \cdot r' \cdot \frac{\Delta\theta}{2 \cdot \pi} = r' \cdot \Delta\theta \quad (4.33)$$

kann $\Delta\theta$ in Abhängigkeit von r' und b beschrieben werden:

$$\Delta\theta = \frac{b}{r'} \quad . \quad (4.34)$$

Somit ergibt sich für die Anzahl der Kreissektoren n_{seg} in Abhängigkeit von r' und b :

$$n_{seg} = \text{ceil}\left(\frac{2\pi}{\frac{b}{r'}}\right) = \text{ceil}\left(\frac{2\pi \cdot r'}{b}\right) \quad . \quad (4.35)$$

Um sicherzustellen, dass ein Kreis trotz sehr kleinen Radius in genug Sektoren aufgeteilt wird, wird falls $r' < r'_{min} = 1$, $r' = r'_{min}$ gesetzt. Somit folgt für $r' < 1$:

$$n_{seg} = \text{ceil}\left(\frac{2\pi}{b}\right) . \quad (4.36)$$

Ein Sektor s_i gilt als positiv sobald ein Punkt aus \mathbf{P}_{circ} auf diesem liegt [10]. Für eine erfolgreiche Validierung müssen mindestens $p_{seg}\%$ der Sektoren positiv sein [10].

Ist die Anzahl der positiven Sektoren zu gering, so wird die Validierung mit einem negativen Ergebnis abgebrochen. Um eine weitere Validierung des Kreises mit den Parametern (u'_m, v'_m, r') zu vermeiden, wird der dazu korrespondierende Eintrag im Akkumulator als validiert markiert, der Kreis wird jedoch nicht der Liste der gefundenen Kreise hinzugefügt.

Eine Validierung ist positiv, sobald mindestens $p_{seg}\%$ der Sektoren positiv sind. Der zuvor grob bestimmte Kreis mit den Parametern (u'_m, v'_m, r') wird mithilfe eines Least-Square Ansatzes (beschrieben in Abschnitt 4.2.5) bestmöglich auf die Punkte \mathbf{P}_{circ} gefittet. Anschließend werden alle Einträge die den neu berechneten Kreis c_{LS} schneiden aus dem Akkumulator entfernt, da dort in den meisten Fällen viele Einträge vorhanden sind, welche dem detektierten Kreis sehr ähnlich sind [10].

Für die Auswertung in Kapitel 5 wird $b = 0.015$ und $p_{seg} = 50\%$ für die Detektion von Vollzylinder und $p_{seg} = 25\%$ für die Detektion von Halbzylindern verwendet. In Abschnitt 5.3 wird zudem gezeigt, dass ein zu klein gewähltes p_{seg} zu einer rasch ansteigenden Falschdetektionsrate führen kann. Dies gilt analog auch für ein zu groß gewähltes b . Gleichzeitig führt ein zu hoch gewähltes p_{seg} oder ein zu klein gesetztes b dazu das Zylinder mit nur wenigen Punkten auf der Mantelfläche nicht validiert werden können (siehe für Beispiel hierfür ebenfalls Abschnitt 5.3).

4.2.5 Least-Square Kreis Fitting

Um anschließend die validierten, grob detektierten Kreise der RHT möglichst gut in die nähere Punkteumgebung zu legen wird ein Least-Square verwendet. Hierfür wurde der Ansatz beschrieben in [25] übernommen. Als Eingabepunkteset für den Algorithmus dienen die zweidimensionalen Punkte \mathbf{P}_{circ} , selektiert aus \mathbf{P}_{2D} . Sie definieren den Kreis \mathbf{K} mit dem Mittelpunkt (\bar{x}, \bar{y}) und dem Radius R . Wobei im Folgenden $\Lambda = [(x_i - \bar{x})^2 + (y_i - \bar{y})^2]$ gegeben sei. Der Fehler wird definiert als die Differenz zwischen der konstanten Fläche πR^2 und der Fläche des Kreises mit dem Radius R zentriert bei \bar{x} und \bar{y} :

$$\sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2} = \sqrt{\Lambda} , \quad (4.37)$$

wodurch der aufsummierte Fehler gegeben ist durch:

$$e(R, \bar{x}, \bar{y}) = \sum_{i=1}^N [\pi R^2 - \pi \Lambda]^2 \quad (4.38)$$

oder

$$J = \frac{e}{\pi^2} = \sum_{i=1}^N [R^2 - \Lambda]^2 \quad . \quad (4.39)$$

Für einen Bestmöglichen Fit muss $J(R, \bar{x}, \bar{y})$ minimiert werden. Diese Minimierung kann mithilfe der partiellen Ableitungen berechnet werden. Differenziert man J partiell nach R so ergibt sich:

$$\frac{\delta J}{\delta R} = 2 \sum_{i=1}^N [R^2 - \Lambda](2R) = 0 \quad , \quad (4.40)$$

woraus sich wiederum ergibt:

$$NR^2 = \sum_{i=1}^N \Lambda \quad . \quad (4.41)$$

Die partielle Ableitung nach \bar{x} liefert:

$$\frac{\delta J}{\delta \bar{x}} = 2 \sum_{i=1}^N [R^2 - \Lambda] 2(x_i - \bar{x})(-1) = 0 \quad (4.42)$$

oder

$$\sum_{i=1}^N [R^2 - \Lambda] x_i = \sum_{i=1}^N [R^2 - \Lambda] \bar{x} = 0 \quad , \quad (4.43)$$

woraus folgt:

$$R^2 \sum_{i=1}^N x_i = \sum_{i=1}^N x_i \quad . \quad (4.44)$$

Analog dazu gilt für die partielle Ableitung für \bar{y} :

$$R^2 \sum_{i=1}^N y_i = \sum_{i=1}^N y_i \quad . \quad (4.45)$$

Angenommen die \mathbf{P}_{circ} seien fehlerfrei, so gilt für jeden Punkt $(\tilde{x}_i, \tilde{y}_i)$ auf dem Kreis die Gleichung:

$$\Lambda = R^2 \quad . \quad (4.46)$$

Diese Aufteilungen besitzen den Vorteil, dass sie trotz ihrer quadratischen Natur gelöst werden können. Hierfür sei die Notation eingeführt, dass $\sum_x = \sum_{i=1}^N x_i$, $\sum_{x^2} = \sum_{i=1}^N x_i^2$, $\sum_{x^3} = \sum_{i=1}^N x_i^3$, $\sum_{xy} = \sum_{i=1}^N x_i y_i$ usw. Somit ergibt sich aus Gleichung 4.41:

$$NR^2 = \sum_{x^2} - 2 \sum_x \bar{x} + N\bar{x}^2 + \sum_{y^2} - 2 \sum_y \bar{y} + N\bar{y}^2 \quad . \quad (4.47)$$

Aus Gleichung 4.44:

$$R^2 \sum_x = \sum_{x^3} - 2 \sum_{x^2} \bar{x} + \sum_x \bar{x}^2 + \sum_{xy^2} - 2 \sum_{xy} \bar{y} + \sum_x \bar{y}^2 \quad (4.48)$$

und analog für Gleichung 4.45:

$$R^2 \sum_y = \sum_{x^2 y} - 2 \sum_{xy} \bar{x} + \sum_{y^3} - 2 \sum_{y^2} \bar{y} + \sum_y \bar{y}^2 \quad . \quad (4.49)$$

Wird Gleichung 4.47 nun mit \sum_x multipliziert und subtrahiert man davon Gleichung 4.44 multipliziert mit N so ergibt sich:

$$\sum_{x^2} \sum_x - N \sum_{x^3} - 2 \bar{x} [\sum_x^2 - N \sum_{x^2}] + \sum_x \sum_{y^2} - N \sum_{xy^2} - 2 \bar{y} [\sum_x \sum_y - N \sum_{yx}] = 0 \quad . \quad (4.50)$$

Analog kann man auch 4.47 mit \sum_y multiplizieren und N Mal 4.44 subtrahieren:

$$\sum_{x^2} \sum_y - N \sum_{x^2 y} - 2 \bar{x} [\sum_x \sum_y - N \sum_{yx}] + \sum_y \sum_{y^2} - N \sum_{y^3} - 2 \bar{y} [\sum_y^2 - N \sum_{y^2}] = 0 \quad . \quad (4.51)$$

Aus 4.50 und 4.51 ergibt sich die Lösung für \bar{x} und \bar{y} :

$$\begin{pmatrix} a_1 & b_1 \\ a_2 & b_2 \end{pmatrix} \begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \quad , \quad (4.52)$$

wobei

$$\begin{aligned} a_1 &= 2(\sum_x^2 - N \sum_{x^2}) & b_1 &= 2(\sum_x \sum_y - N \sum_{xy}) \\ a_2 &= 2(\sum_x \sum_y - N \sum_{xy}) = b_1 & b_2 &= 2(\sum_y^2 - N \sum_{y^2}) \end{aligned} \quad (4.53)$$

$$\begin{aligned} c_1 &= (\sum_{x^2} \sum_x - N \sum_{x^3} + \sum_x \sum_{y^2} - N \sum_{xy^2}) \\ c_2 &= (\sum_{x^2} \sum_y - N \sum_{x^2 y} + \sum_y \sum_{y^2} - N \sum_{x^2 y}) \end{aligned} \quad . \quad (4.54)$$

Daraus folgt die Lösung des Kreismittelpunktes:

$$\bar{x} = \frac{c_1 b_2 - c_2 b_1}{a_1 b_2 - a_2 b_1} \quad (4.55)$$

$$\bar{y} = \frac{a_1 c_2 - a_2 c_1}{a_1 b_2 - a_2 b_1} \quad . \quad (4.56)$$

Mit diesen ist es möglich mithilfe einer Substitution in 4.47 R^2 zu berechnen:

$$R^2 = \frac{1}{N} (\sum_{x^2} - 2 \sum_x \bar{x} + N \bar{x}^2 - 2 \sum_y \bar{y} + N \bar{y}^2) \quad . \quad (4.57)$$

So kann aus dem grob detektierten, validierten Kreis der RHT und den Punkten aus seiner nächsten Umgebung ein möglichst guter Kreisfit berechnet werden. [25]

Auch wenn ein solcher Least-Square Ansatz theoretisch nicht nötig ist, da durch eine genauere RHT eine bessere Hypothese für die Kreisdetektion abgegeben werden kann, so kann mit der Kombination dieser beiden Algorithmen Zeit gespart und gleichzeitig eine gute Kreisschätzung berechnet werden. Somit trägt ein Least-Square Ansatz zu einem besser gefitteten Zylinder bei.

4.2.6 Pseudocode der RHT-Kreisdetektion

In Pseudocode 4 werden noch einmal die wichtigsten Schritte der Kreisdetektion zusammengefasst. Der Pseudocode dient für eine grobe Übersicht der Kreisdetektion, die Details der einzelnen Funktionen, etwa der Validierung werden in den obigen Abschnitten besprochen.

Algorithm 4 Kreisdetektion - RHT

Require: Eingabe: \mathbf{P}_{2D} , mögliche Zylinderpunkte projiziert entlang Zylinderachse

```

1: Akkumulator  $\mathbf{A} = \{\}$ 
2: Kreisliste  $\mathbf{VK} = \{\}$ 
3: while  $t < n$  do
4:   Wähle  $p_1, p_2, p_3 \in \mathbf{P}_{2D}$  zufällig
5:   if  $\text{Kollinear}(p_1, p_2, p_3)$  then
6:      $t++$ ;
7:     continue;
8:   else
9:      $t++$ ;
10:    Kreis  $\mathbf{K} = \text{berechneKreisAus3Punkten}(p_1, p_2, p_3)$ 
11:    if  $\mathbf{A}.\text{hasEintrag}(\mathbf{K})$  then
12:      Akkumulatorzelle  $\mathbf{Z} = \mathbf{A}.\text{getEintrag}(\mathbf{K})$ 
13:       $\mathbf{Z}.\text{counter} += 1$ 
14:      if  $\mathbf{Z}.\text{counter} \geq v$  and  $\mathbf{Z}.\text{validated} == \text{false}$  then
15:         $\text{isCircle} = \text{validiere}(\mathbf{Z}.\mathbf{K})$ 
16:         $\mathbf{Z}.\text{validated} = \text{true}$ 
17:        if  $\text{isCircle}$  then
18:           $\mathbf{K}_{LS} = \text{LeastSquare}(\mathbf{Z}.\mathbf{K})$ 
19:           $\mathbf{VK}.\text{add}(\mathbf{K}_{LS})$ 
20:           $\mathbf{A}.\text{deleteGeschnitteneKreise}(\mathbf{K}_{LS})$ 
21:           $t = 0$ ;
22:        end if
23:      end if
24:    else
25:       $\mathbf{A}.\text{add}(\mathbf{K}, \text{counter}=1, \text{validated} = \text{false})$ 
26:    end if
27:  end if
28: end while
29: return  $\mathbf{VK}$ 

```

4.3 Zylinderenddetektion und Validierung detektierter Zylinder

Mithilfe der Zylinderachsendetektion und der Kreisdetektion ist es möglich alle 5 Zylinderparameter zu bestimmen, wobei die Position des Zylinder nur gegeben ist durch einen Punkt auf der Achse. Um anschließend die Zylinderlänge zu berechnen wird in [22] keine Lösung aufgezeigt. In dieser Arbeit wird der Punkt auf der Zylinderachse genutzt, um entlang der Achse den Anfangs- sowie Endpunkt der Achse ausfindig zu machen. Es werden dazu nicht die gefundenen Kreispunkte genutzt, da zwei Zylinder mit gleicher Achse und Radius, nur getrennt durch eine Translation entlang der Achse denselben projizierten Kreis besitzen.

Nachdem der Algorithmus die letzten Zylinderenden berechnet hat, müssen anschließend alle gefundenen Zylinder miteinander verglichen und doppelte entfernt werden. Dies ist in [22] nicht nötig, da für jede gefundene Zylinderachse eine eigene Kreisdetektion berechnet wird, die nur ein Maximum auswertet und anschließend die Zylinderpunkte aus der Punktwolke löscht. Da in dem hier implementierten Fall keine Punkte aus der Punktwolke gelöscht werden, ist es möglich Mehrfachdetektionen zu finden. Diese Detektionen müssen erkannt und schließlich ausgewertet werden, sodass nur der bestmögliche Fit am Ende der Zylinderdetektion zurückgegeben wird. In den folgenden Abschnitten wird aus diesem Grund zuerst die Zylinderenddetektion (siehe 4.3.1) und anschließend die Erkennung von doppelten Zylindern (siehe 4.3.2) erklärt.

4.3.1 Zylinderenddetektion

Die Zylinderenddetektion erfolgt direkt nach der Kreisdetektion. Dies hat den Vorteil, dass nicht die komplette Punktwolke betrachtet werden muss, sondern nur das bereits für die Kreisdetektion vorselektierte Punkteset \mathbf{P}_c , deren sphärische Normalen für die Zylinderachse voten. Es werden nicht direkt die Punkte eines Kreises genutzt, da ein Kreis auch zwei Zylinder mit gleicher Achse und Radius repräsentieren kann.

Um aus \mathbf{P}_c alle Achsenend- beziehungsweise Anfangspunkte für die Zylinderachse \mathbf{w} zu bestimmen, müssen für jeden Zylinder \mathbf{Z} alle Punkte aus \mathbf{P}_{cyl} selektiert werden, die auf der Zylindermantelfläche liegen. Hierfür wird aus \mathbf{w} und dem Punkt auf der Achse \mathbf{p}_w , gegeben durch den transformierten Mittelpunkt des Zylinderkreises \mathbf{K} , eine Gerade \mathbf{g} definiert und für jeden Punkt \mathbf{p}_{PC} aus \mathbf{P}_c die Distanz d zu \mathbf{g} berechnet. Alle N Punkte, für die gilt $r - \lambda \leq d \leq r + \lambda$, wobei r der Radius von \mathbf{K} und λ ein kleiner Schwellwert, bilden \mathbf{P}_{cyl} .

Anschließend werden alle Punkte aus \mathbf{P}_{cyl} senkrecht auf \mathbf{g} projiziert und sortiert, sodass \mathbf{p}_1 und \mathbf{p}_N die größte Distanz zwischen zwei Punkten in \mathbf{P}_{cyl} projiziert auf \mathbf{g} besitzen, wobei \mathbf{p}_1 und $\mathbf{p}_N \in \mathbf{P}_{cyl}$. Anschließend wird die Distanz d_n zwischen jeden projizierten Punkt \mathbf{p}_i und seinen Nachbarn \mathbf{p}_{i+1} beziehungsweise \mathbf{p}_{i-1} , falls vorhanden, gebildet. Aus allen Distanzen d_n wird anschließend der Mittelwert d_{mean} gebildet. Da Aufgrund der Sortierung die Punkte entlang \mathbf{g} angeordnet sind, gibt d_{mean} eine grobe Auskunft über die durchschnittliche Distanz zwischen zwei Punkten auf der Gerade aus.

Aus diesem Grund kann d_{mean} dazu verwendet werden, um alle projizierten Punkte in Gruppen aufzuteilen, wobei die Distanz zwischen jedem Punkt \mathbf{p}_i aus einer beliebigen Gruppe \mathbf{G}_k und

seinen möglichen Nachbarn \mathbf{p}_{i+1} , \mathbf{p}_{i-1} kleiner gleich $d_{mean} \cdot t_m$ sein muss. Dabei gibt der mit d_{mean} multiplizierte Faktor t_m die Möglichkeit die mittlere Distanz d_{mean} anzupassen. Die Gruppe mit den meisten Elementen wird als Mantelfläche des Zylinders bezeichnet \mathbf{P}_{MZ} . Aufgrund der Vorsortierung kann aus dem Mittelwert der i ersten und letzten Punkte direkt der Zylinderachsenanfang beziehungsweise das Zylinderachsenende bestimmt werden.

Der Vorteil dieses Vorgehens liegt in der einfachen Implementierung, sowie in dem bereits vorselektierten Punkteset \mathbf{P}_c , das meistens deutlich kleiner als die Punktwolke ist. Gleichzeitig bringt jedoch letzteres den Nachteil, dass nicht alle Zylinderpunkte in \mathbf{P}_c vorhanden sind. Gerade die sphärischen Normalen der Punkte am Zylindermantelrand weisen oft eine zu große Differenz im Vergleich zu den restlichen Mantelpunkten auf, sodass sie nicht für die Ursprungsebene mit dem Ebenenvektor gleich der Zylinderachse voten. Aus diesem Grund weist die berechnete Länge des Zylinders, definiert durch die Zylinderachsenendpunkte, stets einen systematisch negativen Fehler auf. Sollte ein solcher Fehler kritisch sein, muss die Zylinderendberechnung angepasst oder eine Verbesserung der Zylinderenden nach Rückgabe des Algorithmus implementiert werden. Die Zylindermantelpunkte können etwa durch eine Punktwolkensuche entlang der Zylinderachse gefunden werden, durch eine anschließende Projektion der Punkte auf die Zylinderachse können neue Maxima beziehungsweise Minima für Zylinderende und Zylinderstart definiert werden.

4.3.2 Erkennung von doppelten Zylindern

Der letzte Schritt der Zylinderdetektion ist eine Validierung der detektierten Zylinder, dabei sollen mehrfach gefundene Zylinder erkannt und die schlechteren Fits anschließend gelöscht werden. Dies ist notwendig, da mehrere ähnliche Zylinderachsenhypothesen berechnet werden können. Was dazu führen kann, dass Zylinder mehrfach auf eine Ebene projiziert und als fehlerbehaftete Kreis detektiert werden können. Ein beliebiger Zylinder \mathbf{Z}_i wird mit einem anderen Zylinder \mathbf{Z}_j nur dann verglichen, wenn der Winkel zwischen beiden Zylinderachsen \mathbf{w}_i und \mathbf{w}_j kleiner als t_w ist. Für die Experimente in Kapitel 5 wurde t_w kleiner 5° gewählt. Ist zudem der minimale Abstand d_{min} zwischen den zwei Linien beschrieben durch \mathbf{w}_i , dem Zylinderachsenstartpunkt \mathbf{p}_{s_i} und Endpunkt \mathbf{p}_{e_i} beziehungsweise \mathbf{w}_j , \mathbf{p}_{s_j} und \mathbf{p}_{e_j} kleiner als die summierten Radii r_i , r_j so schneiden sich die zwei Zylinder und gelten als gleiche Detektion. Sollten viele Zylinder mit ähnlicher oder gleicher Achse sehr nahe beieinander liegen, so sollte die geforderte minimal Distanz d_{min} der Zylinder durch $r_i + r_j - t_{dis}$ ersetzt werden, um das herausfiltern von zwei Nachbarzylinder zu vermeiden. Dies ist für die Experimente beschrieben in Kapitel 5 nicht notwendig.

Nach der Erkennung zweier gleicher Zylinder, gilt es den wahrscheinlich schlechteren aus dem gefundenen Zylinderset \mathbf{Z} zu entfernen. Hierfür werden die Punkte der Zylinder auf dem Zylindermantel \mathbf{P}_{MZ} verglichen und der Zylinder mit der kleineren Anzahl an Mantelpunkten gelöscht. Sollten zwei Zylinder dieselbe Anzahl an Mantelpunkten besitzen, so wird der Zylinder mit dem größeren Radius gelöscht. Bei einer genaueren Zylinderlängenbestimmung, etwa mit allen Punkten, sollte hier der Radius durch die Mantelfläche ersetzt werden.

Anschließend werden alle detektierten Zylinder mit der Zylinderachse, dem Radius, dem berechneten Punkt auf der Zylinderachse sowie die daraus berechneten Zylinderachsenenden

dem Benutzer zurückgegeben. In den nächsten Abschnitten wird die Zylinderdetektion mithilfe von Simulations- sowie Realpunktwolken ausgewertet und die einzelnen Fehler der 5 RHT Parameter berechnet.

Kapitel 5

Experimente und Auswertung

Wie in Kapitel 4 beschrieben, besteht die Zylinderdetektion insgesamt aus zwei sequentiellen Schritten, der Zylinderachsendetektion und der Kreisdetektion. Neben diesen Hauptschritten wird die Zylinderlänge mithilfe der mittleren Distanz zwischen den Mantelflächenpunkte berechnet, zudem werden am Ende der Detektion gleiche Zylinder erkannt und nur der beste Fit beibehalten. Für die Auswertung dieser Zylinderdetektion wurden drei CAD-Modelle genutzt (für bildliche Darstellung der Modelle siehe Anhang 6), wobei zwei von ihnen für die Aufnahme der realen Daten 3D-gedruckt wurden. Somit sind die Zylinder relativ klein und besitzen etwa einen Radius zwischen 2,5mm und 10 mm.

Für die Auswertung der Zylinderdetektion sind die Parameter bis auf zwei Ausnahmen stets gleich gewählt worden. So wird etwa für die Detektion der Zylinderachsen ein Akkumulator mit 636473 quadratischen Zellen verwendet. Um anschließend die Maxima des Akkumulators zu bestimmen, wurde für die „Sliding-Window“-Funktion ein quadratisches Fenster von 20x20 Zellen genutzt. Die maximale Anzahl der detektierten Zylinderachsen, welche an die Kreisdetektion weiter gereicht wird, ist einer der zwei variablen Parameter. Sie entspricht für alle Experimente der Anzahl der einzigartigen Zylinderachsen des Modells multipliziert mit dem Faktor 10. Für die RHT der Kreisdetektion wurde ein Validierungsschwellwert von 3 genutzt. Wurde ein Kreis dreimal gefunden, so wird getestet ob er mindestens 250 Kreispunkte besitzt und seine Kosten größer gleich 5 sind. Wenn dies der Fall ist, wird beim zweiten Schritt der Validierung der Kreise in Sektoren unterteilt, wobei jeder Sektor eine Kreisbogen Länge von 0,016mm besitzt. Für eine erfolgreiche Validierung müssen mindestens $p_{seg}\%$ Sektoren positiv sein. $p_{seg}\%$ ist der zweite variable Parameter. Für die Detektion der Zylindermodelle mit Vollzylindern wurde $p_{seg}\% = 50\%$ gesetzt, während für die Detektion der Teilzylinder $p_{seg}\% = 25\%$ entspricht. Nach der Zylinderdetektion werden mehrfach detektierte Zylinder gesucht und mögliche schlechtere Detektionen gelöscht, sodass stets nur ein gleicher Zylinder dem Nutzer zurückgegeben wird. Dabei werden zwei detektierte Zylinder als gleich gewertet, sobald zwischen den Zylinderachsen ein Winkel $\theta \leq 5^\circ$ liegt und sich beide Zylinder schneiden.

Während die RHT für die Zylinderachsendetektion der Standard Hough-Transformation, beschrieben in [22], ähnlich ist, wurde für die Kreisdetektion ein anderer Algorithmus gewählt.

Der in dieser Arbeit entwickelte RHT Algorithmus setzt sich aus dem RANSAC Ansatz des Papers [15], kombiniert mit der RHT Grundidee beschrieben in [10] und einer Least-Square Optimierung auf der Grundlage von [25] zusammen. Aufgrund der Kombination von drei Ansätzen wird im Folgenden zuerst anhand eines einfachen simulierten Beispiels demonstriert, dass die Grundfunktionalität der Kreisdetektion gegeben ist, bevor die eigentliche Validierung der Zylinderdetektion und anschließend der Teilzylinderdetektion auf simulierten sowie realen Daten erfolgt.

5.1 Auswertung der Kreisdetektion

Um möglichst effizient die Grundfunktionalität der Kreisdetektion anhand eines Zylinderbeispiels zu testen, wird die maximale Anzahl der Zylinderachsen auf eins minimiert. So ist es möglich die detektierten Kreismittelpunkte direkt mit absoluten Werten zu vergleichen und gleichzeitig eine Auswahl an verschiedenen Radii sowie Zylinderhöhen zu betrachten.

Aus diesem Grund werden für die Auswertung der Kreisdetektion insgesamt 16 Zylinder betrachtet, die alle senkrecht in einer 4x4 Matrix auf einer Bodenplatte stehen. Zwischen jedem Zylinder und dessen Nachbarn befindet sich ein Abstand von 30mm. Es werden insgesamt vier verschiedene Radii (2,5mm, 5mm, 7,5mm, 10 mm) und vier verschiedenen Höhen (10mm, 15mm, 20mm, 25mm) mithilfe der CAD-Software Onshape [3] modelliert. In Abbildung 6.1 wird das Modell aus verschiedenen Blickwinkeln und mit den wichtigsten Abmessungen dargestellt.

Das 3D-Modell wird für die Zylinderdetektion anschließend als Mesh exportiert und mithilfe von CloudCompare [2] gesampelt. Der Vorteil einer solchen Punktwolke ist, dass die absoluten Positionen der einzelnen Zylinder in der Punktwolke bekannt sind. Zwar wurde für die Modellierung eine Bodenplatte genutzt, jedoch muss diese aus der Punktwolke entfernt werden, da Ebenen eine mögliche Zylinderdetektion erschweren oder unmöglich gestalten [22]. Eine mögliche Lösung für dieses Problem ist etwa eine vorgeschaltete Ebenendetektion. Im Abschnitt zur Auswertung der Zylinderdetektion (5.2) wird auf das Problem von Ebenenpunkten und wie sie die Zylinderdetektion erschweren beziehungsweise verhindern können noch einmal genauer eingegangen. Die Punktwolke zum Testen der Kreisdetektion besteht nach dem Entfernen der Ebene aus 44772 Punkten. Das Berechnen der Normalen, welche für die Zylinderdetektion von Nöten sind, sowie das Anzeigen von Punktwolken, Hough-Räumen und Zylindern erfolgt mithilfe des Toolkit 3DTK [1].

Da für die Kreisdetektion ein randomisiertes Verfahren verwendet wird, wurde für die Auswertung das Ergebnis von 1000 Detektionen betrachtet. Somit sollte in einem Testdatensatz jeder Zylinder 1000 Mal detektiert werden und im Optimalfall 16000 Zylinder gefunden werden. In dem speziellen Fall der Kreisdetektionsvalidierung wurden insgesamt zwei Testdatensätze erzeugt. Im ersten Versuch wurde der Kreisdetektion eine perfekte Zylinderachse vorgegeben, während im zweiten Versuch eine detektierte und somit fehlerbehaftete Zylinderachse übergeben wird. Diese Aufteilung ist für eine Erstauswertung notwendig, da eine fehlerbehaftete Achse zu einem systematischen Fehler in der Projektion führt, welcher nicht durch den Least-Square Ansatz behoben werden kann.

5.1.1 Auswertung Kreisdetektion bei fehlerfreier Zylinderachse und rauschfreier Punktwolke

Zur Auswertung der Grundfunktionalität der Kreisdetektion, ist eine Berechnung der Zylinderachsendetektion nicht notwendig, stattdessen wird die Zylinderachse direkt der Kreisdetektion übergeben. Im Falle eines rechtshändischen Koordinatensystems wäre die Achse $[0\ 0\ 1]$, da 3DTK jedoch standardmäßig linkshändische Koordinatensysteme verwendet, ist die Zylinderachse $[0\ -1\ 0]$, beziehungsweise aufgrund der möglichen Invertierung $[0\ 1\ 0]$. Ein solcher Test der Grundfunktionalität ist notwendig um zu beobachten, ob der Algorithmus kombiniert aus den Ansätzen der Paper [15], [10] und [25] fehlerfrei und robust funktioniert oder ob er einen systematischen Fehler aufweist. Bei der Verwendung von fehlerbehafteten detektierten Achsen kann keine direkte Aussage über die Funktionalität der Kreisdetektion getroffen werden, da sich der Fehler, durch die Projektion der 3D-Punkte entlang der Zylinderachse auf eine Ebene, fortpflanzt.

Betrachtet man die Ergebnisse der Kreisdetektion, so wurden bei einer vorgegebenen Achse und 1000 randomisierten Versuchen, alle Zylinder exakt 1000 Mal gefunden. Folglich wurden alle Kreise zumindest detektiert. Aus diesem Grund wird in den folgenden Abschnitten ein genaues Augenmerk auf die Fehler der Radii in Relation zur Größe geworfen und anschließend die absoluten Positionen der Kreismittelpunkte und deren Fehler analysiert.

Fehler der Radiusbestimmung

Zu Beginn der Auswertung werden alle gefundenen Zylinder sortiert, so dass die 16000 Detektionen, in je 16 Mal 1000 detektierte Kreise aufgeteilt sind. Anschließend werden zuerst die gefundenen Radii für jeden Zylinder einzeln betrachtet. Im Modell existieren vier verschiedene Radiigrößen mit 2,5mm, 5,0mm, 7,5mm und 10,0mm. Jeder Radius kommt exakt vier Mal in dem Modell vor, wobei kein Zylinder mit dem gleichen Radius dieselbe Höhe aufweist. Somit müssen bei 1000 Durchläufen insgesamt 4000 Kreise für jede Radiusgröße gefunden werden. In Abbildung 5.1 werden alle berechneten Radii in einem Histogramm dargestellt. Man kann gut erkennen, dass vier Peaks bei 2,5mm, 5,0mm, 7,5mm und 10mm vorhanden sind. Jeder dieser Radii wurde exakt 4000 Mal detektiert.

Aufgrund der geringen Unterschiede zwischen den einzelnen detektierten Radii, ist es nicht möglich diese in einem Histogramm mit einer Radius Skala von 2 bis 10 mm darzustellen, deshalb werden im Folgenden immer die Fehler der Radii in Millimeter betrachtet. In den vier Histogrammen dargestellt in Abbildung 5.2, werden die Fehler der Radii aufgeschlüsselt nach der Radiusgröße in jeweils einem Histogramm gezeigt. Auffällig ist dabei, dass in den Teilhistogrammen für jeden Radius exakt vier Maxima, mit Ausnahme des Histogramms für den Radius von 7,5mm, erkennbar sind. Betrachtet wird im Folgenden als Beispiel das Histogramm für den Radius 2,5mm, dargestellt in Abbildung 5.2(a). Die Maxima der Fehler liegen in diesem Fall bei $-0,00041$, $-0,0004$, $-0,00039$ und $-0,00036$ mm. Jede Abweichung wird exakt 1000 Mal detektiert. Eine ähnliche Maxima Verteilung liegt auch bei einem Radius von 5,0mm und 10mm vor. Die einzige Ausnahme bilden die Fehler des Radius 7,5mm, dargestellt in 5.2(c), hier werden

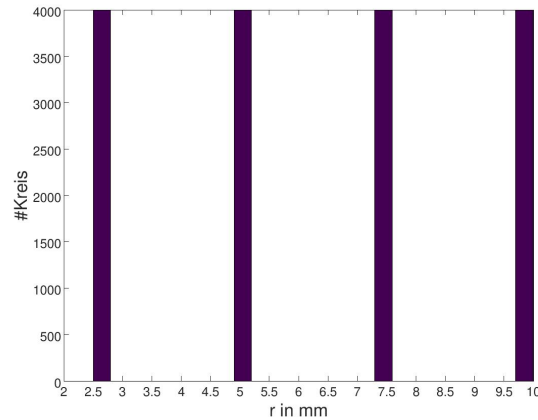


Abbildung 5.1: Histogramm der gefundenen Radii

insgesamt nur drei Peaks gefunden, jedoch besitzt der erste Peak bei $-0,0119\text{mm}$ die doppelte Höhe. Im Falle des Radius $7,5\text{mm}$ fallen also zwei Fehlerpeaks zusammen.

Eine Erklärung für diese deterministische Verteilung findet sich in der Aufbauweise des Modells wieder. Zwar besitzen exakt vier Zylinder immer denselben Radius, jedoch ist ihre Höhe und Position stets verschieden. Die Unterschiede in der Höhe und somit in der Anzahl und Aufteilung der Mantelflächenpunkte zeigt sich in der Detektion der Kreise wieder. Aufgrund der RHT scheint ein derart deterministisches Verhalten auf den ersten Blick ungewöhnlich. Da jedoch nach einer erfolgreichen Kreisdetektion, ein Least-Square Ansatz über alle Punkte in nächster Kreisumgebung ($r_{min} = r_{rht} - 1, r_{max} = r_{rht} + 1$) berechnet wird, erfolgt nach der randomisierten Detektion ein deterministisches Fitting in kleiner Umgebung. Dieser kombinierte Ansatz ermöglicht es trotz sehr schneller grober RHT einen sehr guten Kreis zuverlässig zu detektieren. Diese deterministische Verbesserung führt zudem auch zu den vier klar erkennbaren Peaks in den Histogrammen. Dabei ist der Fehlerbereich der einzelnen Radii so gering, dass eindeutig eine erfolgreiche Radiusbestimmung vorliegt.

Radius [mm]	Mittelwert Radius [mm]	Mittlerer Fehler [%] in Abhängigkeit des Radius	Standard- abweichung [mm]
2,5	2,4996	-0,0156	1,8711e-05
5,0	4,9992	-0,0155	1,4792e-05
7,5	7,4988	-0,0157	1,2992e-05
10,0	9,9984	-0,0159	1,4792e-05

Tabelle 5.1: Mittlerer Fehler und Standardabweichung der vier Radii

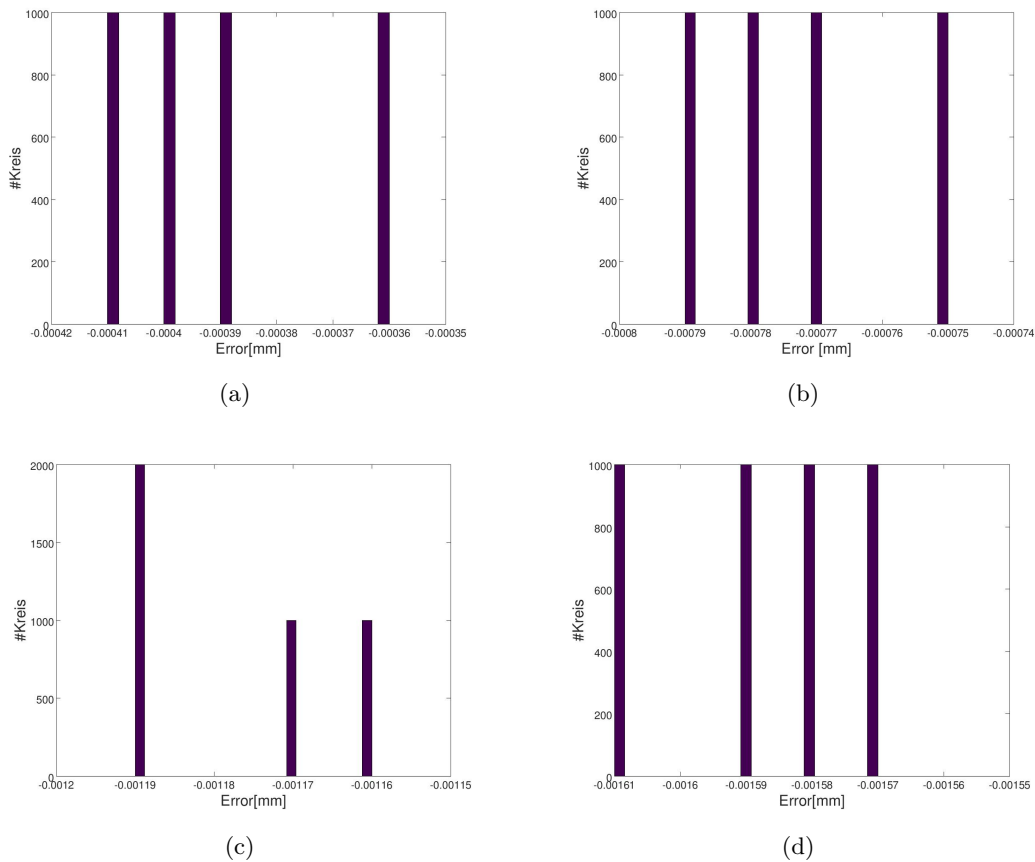


Abbildung 5.2: Fehler der vier Radii:(a) Fehler der Kreise mit Radius 2,5mm;(b) Fehler der Kreise mit Radius 5,0mm;(c) Fehler der Kreise mit Radius 7,5mm; (d) Fehler der Kreise mit Radius 10mm.

Wird etwa der Fehlerbereich des Radius 2,5mm betrachtet, so liegt dieser im Bereich von $-0,00036\text{mm}$ bis $-0,00041\text{mm}$, während die Fehler der Kreise mit Radius 10,0mm sich im Bereich zwischen $-0,00157\text{mm}$ und $-0,00161\text{mm}$ befinden. Wird der mittlere Fehler für die einzelnen Radii berechnet, so kann an den Werten $-3,900\text{e-}04$, $-7,725\text{e-}04$, $-11,775\text{e-}04$ und $15,875\text{e-}04$ gut erkannt werden, dass der mittlere Fehler mit steigendem Radius zunimmt. In Tabelle 5.1 wird neben den mittleren Fehlern angeordnet nach steigendem Radius auch die Standardabweichung sowie der mittlere Fehler in Abhängigkeit vom Radius angezeigt. Dabei fällt auf, dass bei allen vier Radii dieser Wert bei etwa $-0,016\%$ liegt.

Des Weiteren ist jedoch anzumerken, dass alle gefundenen Radii minimal kleiner sind als der absolute Wert und somit stets einen negativen Fehler besitzen. Hierfür kann es zwei mögliche Fehlerquellen geben. Zum einen kann der systematisch negative Fehler auf das Least-Square Fitting am Ende einer erfolgreichen RHT zurückgeführt werden. Zum anderen ist es möglich, dass diese minimale Abweichung tatsächlich existiert, da die Punktwolke durch das Samplen eines 3D-Modells entsteht. Zylinder sowie Kreise müssen in CAD Programmen durch Polygone

angenähert werden, aufgrund dieser notwendigen Approximation kann es ebenfalls zu dieser minimalen Abweichung kommen. Im Folgenden wird dieser Fehler aufgrund seiner geringen Größe und seinem nicht sichtbaren Einfluss vernachlässigt.

Zusammengefasst kann man für die Auswertung der Radiusdetektion bei einer exakten Zylinderachse feststellen, dass alle Radii gefunden und in einem sehr guten Fehlerbereich liegen, wobei die Streuweite der Daten sehr klein ist. Einzig und allein der sehr kleine, systematisch negative Fehler des Radius ist anzumerken. Diese Abweichungen sind im Vergleich zu den späteren Fehlern der realen Daten oder der durch Ebenenpunkte erzeugten Ungenauigkeiten jedoch so klein, dass sie im späteren Verlauf keinen Einfluss zeigen. Insgesamt kann festgehalten werden, dass der Zylinderparameter Radius stets sehr genau detektiert wurde.

Fehler der Positionsbestimmung

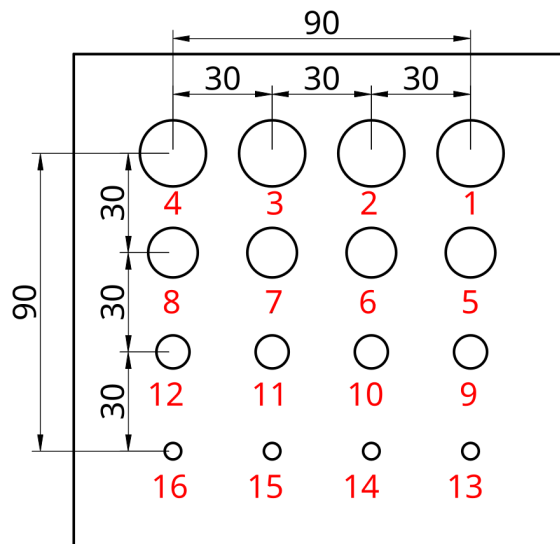


Abbildung 5.3: Darstellung der Zylindernummern (rot) im Modell 1

Neben dem Radius wird durch die Kreisdetektion auch die Zylinderposition im dreidimensionalen Raum bestimmt. Dies erfolgt über die Transformation des projizierten Kreismittelpunktes zurück in den dreidimensionalen Raum. Für die Auswertung des Achsenpunktes wird der 3D-Punkt zurück auf die Bodenplatte projiziert, sodass er direkt mit den absoluten Kreismittelpunktpositionen verglichen werden kann. Dies ist nur möglich, da es sich in diesem speziellen Fall, um einen simulierten Datensatz mit bekannter Zylinderachse handelt.

Für die absoluten Positionen wurde der Koordinatenursprung $[0,0]$, der projizierten Ebenenachsen u und v , in die Mitte der Bodenplatte gelegt. Die Zylinder stehen alle senkrecht auf dieser, sodass die Mantelpunkte eines Zylinders projiziert entlang ihrer Achse einen Kreis auf der Ebene bilden. Der absolute Kreismittelpunkt des Zylinders mit Radius 10,0mm und Höhe 25mm

liegt bei $[-45, -45]$, während der Mittelpunkt des Zylinders mit Radius 2,5mm und Höhe 10mm bei $[45, 45]$ liegt. Die im folgenden genutzten Zylindernummern werden in der Abbildung 5.3 dargestellt.

Zylinder gegeben durch [Radius, Höhe]	Optimale Position [u,v] [mm]	Absoluter mittlerer Fehler[u,v] [mm]	Distanz Standard- abweichung [mm]
Zyl. 1 [10mm, 25mm]	$[-45 -45]$	$[0,0 0,0]$	0,0
Zyl. 2 [10mm, 20mm]	$[-15 -45]$	$[0,0 0,0]$	0,0
Zyl. 3 [10mm, 15mm]	$[15 -45]$	$[0,0 0,0]$	0,0
Zyl. 4 [10mm, 10mm]	$[45 -45]$	$[0,0 0,0]$	0,0
Zyl. 5 [7,5mm, 25mm]	$[-45, -15]$	$[0,0 0,0]$	0,0
Zyl. 6 [7,5mm, 20mm]	$[-15, -15]$	$[0,0 0,0]$	0,0
Zyl. 7 [7,5mm, 15mm]	$[15, -15]$	$[0,0 0,0]$	0,0
Zyl. 8 [7,5mm, 10mm]	$[45, -15]$	$[0,0 0,0]$	0,0
Zyl. 9 [5,0mm, 25mm]	$[-45, 15]$	$[0,0 0,0]$	0,0
Zyl. 10 [5,0mm, 20mm]	$[-15, 15]$	$[0,0 0,0]$	0,0
Zyl. 11 [5,0mm, 15mm]	$[15, 15]$	$[0,0 0,0]$	0,0
Zyl. 12 [5,0mm, 10mm]	$[45, 15]$	$[0,0 0,0]$	0,0
Zyl. 13 [2,5mm, 25mm]	$[-45, 45]$	$[0,0 0,0]$	0,0
Zyl. 14 [2,5mm, 20mm]	$[-15, 45]$	$[0,0 0,0]$	0,0
Zyl. 15 [2,5mm, 15mm]	$[15, 45]$	$[0,0 0,0]$	0,0
Zyl. 16 [2,5mm, 10mm]	$[45, 45]$	$[0,0 0,0]$	0,0

Tabelle 5.2: Mittlerer Fehler und Standardabweichung der Zylinder Positionen

Wird der absolute Fehler der Positionen, sowie der Standardabweichung betrachtet (siehe Tabelle 5.2), so ist ersichtlich, dass die Kreismittelpunkte stets denselben exakten Wert aufweisen. Aus Gründen der Lesbarkeit wurden in Tabelle 5.2 nur zwei signifikante Stellen, statt der berechneten fünf angegeben, da auch in diesen keine Abweichung ersichtlich ist. Diese exakte Kreismittelpunktbestimmung, trotz schneller ungenauer RHT, ist durch den Least-Square Ansatz möglich. Auf einzelne Histogramme wird aufgrund des Ergebnisses in diesem Fall verzichtet.

Somit kann eindeutig zusammengefasst werden, dass die Kreismittelpunkte bei simulierten Daten und perfekter Achse, sehr genau bestimmt werden können. Auf den ausgewerteten fünf signifikanten Stellen ist kein Fehler zu erkennen.

Auswertung der Radius- und Positionsbestimmung

Insgesamt kann eindeutig festgehalten werden, dass die Kreisdetektion funktionsfähig ist, wenn eine nicht fehlerbehaftete Achse und eine aus einem Mesh gesampelte, also rauschfreie, Punktwolke als Eingabe genutzt wird. Selbst Zylinderkreise mit sehr wenigen Mantelpunkten, dargestellt durch den Zylinder mit Radius 2,5mm und Höhe 10mm, werden stets gefunden. Da eine fehlerfreie Achse jedoch ein absoluter Randfall ist, der in der Realität meist nicht vorkommt, wird im Folgenden Abschnitt die Funktionalität der Kreisdetektion bei einer fehlerbehaftet detektierten Achse getestet. Anschließend wird die komplette Zylinderdetektion anhand zweier Beispiele mit je Simulations- sowie Realdaten ausgewertet.

5.1.2 Auswertung der Kreisdetektion bei fehlerbehafteter Zylinderachse und rauschfreier Punktwolke

In diesem Abschnitt wird die Kreisdetektion auf projizierte Ebenenpunkte angewandt, die aufgrund der detektierten, fehlerbehafteten Zylinderachse einen systematischen Fehler besitzen. Dieser systematische Fehler ist abhängig, von der Größe des Fehlerwinkels zwischen absoluter und detektierter Zylinderachse. Für die Detektion von systematisch fehlerbehafteten Kreisen gilt: Je größer der Fehler und je weniger Mantelpunkte ein Zylinder besitzt, desto schwerer ist es seinen Zylinderkreis zu detektieren.

In der Auswertung dieses Testsets tritt ein Fall auf, woran man genau dies erkennen kann, denn anstatt der 16000 Zylinder wurden nur 15999 Zylinder gefunden. Werden die Detektionen auf die einzelnen Zylinder aufgeteilt, so ist schnell zu erkennen, dass bei 1000 Versuchen ein Zylinder mit Radius 2,5mm und einer Höhe von 10,0mm genau einmal nicht detektiert wurde. Der Grund hierfür ist, dass auf ihm die wenigsten Mantelpunkte liegen, wodurch es, bei einer leicht verschobenen Achse, deutlich schwieriger wird den Zylinder trotzdem zu detektieren. Der entscheidende limitierende Faktor ist dabei nicht die RHT, sondern die Validierungsfunktion. Wie bereits im Kapitel Implementierung (4) beschrieben, teilt die Validierungsfunktion einen Kreis in Sektoren auf, abhängig von einer durch den Benutzer definierten Auflösung und dem Radius. Bei einer Detektion von ganzen Zylindern müssen mindestens 50% aller Sektoren positiv, also einen auf dem Kreisrand liegenden Punkt besitzen. Wird nun für die Projektion eine fehlerbehaftete Zylinderachse benutzt, so kann es aufgrund der ellipsenartigen Projektion dazu kommen, dass zu wenig Punkte im nächsten Umfeld des Kreisrandes liegen. Dies führt zu einer negativen Validierung. Der Kreis wird verworfen und ein weiterer ähnlicher Kreis muss mindestens v -Mal für eine weitere Validierung detektiert werden. Aus diesem Grund kann es bei Zylindern mit kleiner Mantelfläche schneller zu einer Fehldetektion kommen.

Solche Fehldetektionen können durch eine Anpassung der Validierungsparameter minimiert werden. Etwa ist es möglich die für die Sektoraufteilung genutzte Auflösung anzupassen oder auch die minimale Anzahl an positiven Sektoren herabzusetzen, um halbe Zylinder zu detektieren. Eine solche Anpassung führt jedoch zu vermehrten Falschdetektionen, was ebenfalls in der Auswertung der Detektion der halben Zylinder noch genauer beschrieben wird. Da es bei $1000 \cdot 16$ Zylinderdetektionen zu keiner Falschdetektion und nur eine Fehldetektion kommt, sind die Kreisdetektionsparameter gut genug für eine Auswertung der Kreisdetektion gewählt worden.

Detektierte Zylinderachsen

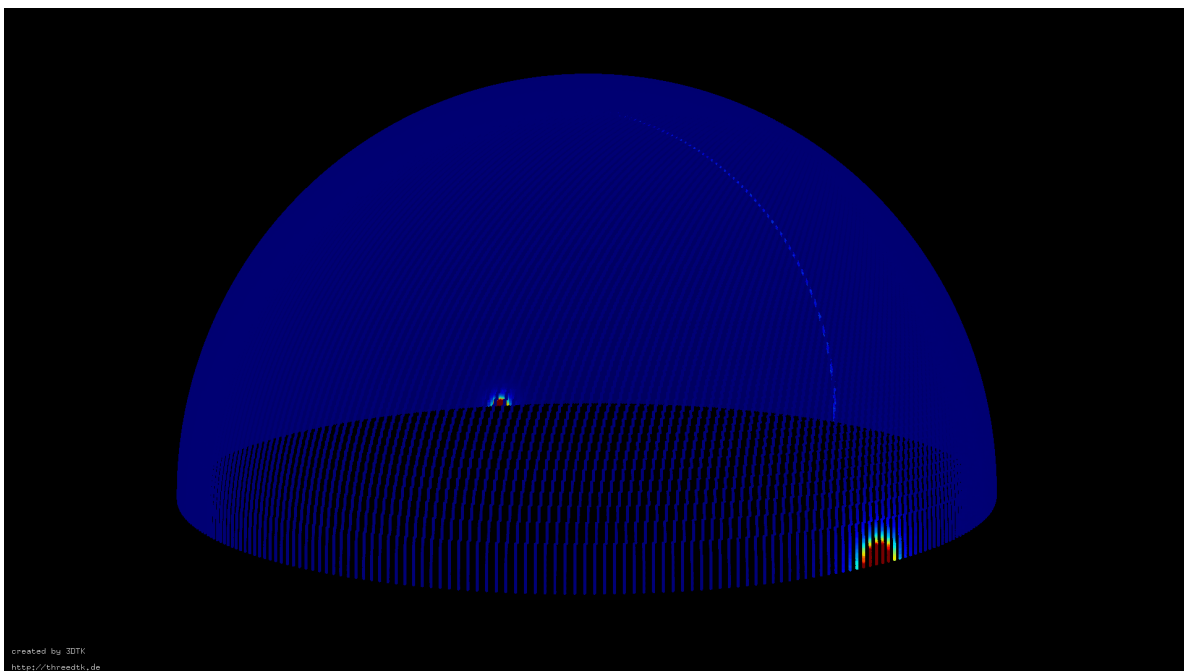


Abbildung 5.4: Darstellung des Akkumulators zur Detektion der Zylinderachsen, wobei blau nicht inkrementiert und rot maximal akkumulierte Zellen beschreibt

Bevor die eigentliche Auswertung der Kreisdetektion beginnt ist es sinnvoll kurz die detektierten Zylinderachsen und ihre Fehler auszuwerten, um zu verstehen welche systematischen Fehler der Kreisdetektion übergeben werden. Dies ist wichtig, da selbst kleine Fehler der Zylinderachsen einen Einfluss auf die Projektion nehmen und somit die Kreisdetektion erschweren können.

Wie bereits zuvor wird als Eingabe eine linkshändige Punktwolke des im Anhang 6.1 dargestellten Modells verwendet. Die perfekte Achse jedes Zylinders ist somit $[0, 1, 0]$ oder $[0, -1, 0]$. Aufgrund dieser Achse werden im Hough-Raum am Rande der Sphäre zwei Maxima berechnet. Der Hough-Raum ist noch einmal graphisch in Abbildung 5.4 dargestellt, wobei blaue Punkte Akkumulatorzellen mit niedrigen Werten und rote Punkte den Maxima entsprechen. Da der Zylinderdetektion nicht bekannt ist wie viele Zylinderachsen in der Punktwolke vorkommen,

können für das Modell trotz eigentlich einheitlicher Achse mehrere Achsen detektiert werden. Ein Abbruch für neue Kreisdetektionen findet erst statt, wenn genug Punkte Zylindern zugewiesen werden, die Maxima im Hough-Raum abgearbeitet sind oder die vom Nutzer vorgegebene maximale Anzahl an Achsen (in diesem Fall 10) überschritten werden.

Detektierte Achse	Anzahl Detektionen	Fehlerwinkeldifferenz[°]
[0,003148 -0,999992 0,002356]	15	0,229183
[0,003148 0,999992 0,002356]	80	0,229183
[0,003148 -0,999995 0,000785]	3328	0,181185
[0,003148 0,999995 0,000785]	12576	0,181185

Tabelle 5.3: Detektierte Zylinderachsen und ihre Fehlerwinkel

In Tabelle 5.3 kann man gut erkennen, dass für das Modell bei 1000 Durchläufen insgesamt vier Zylinderachsen gefunden wurden, wobei jeweils zwei Zylinderachsen ein Paar darstellen, die von den zwei Maxima des Hough-Raumes stammen. Die zwei Zylinderachsen mit einer Fehlerwinkeldifferenz von $0,181^\circ$ werden für etwa 99,4% der Zylinder detektiert, während die zwei Achsen mit der größeren Winkeldifferenz von $0,229^\circ$ gerade einmal für 0,6% der Durchläufe detektiert werden. Alle vier Achsen weisen jedoch nur eine kleine Abweichung im Vergleich zur Realität auf und können etwa durch einen größeren Akkumulator, häufigere zufällige Berechnungen der Großkreise oder mehr Sample Punkten noch an Qualität gewinnen.

Fehler der Radiusbestimmung

Zu Beginn der Auswertung der Kreisdetektion werden erneut zuerst die detektierten Radii betrachtet, anschließend wird die Kreismittelpunktbestimmung validiert. Ein wichtiger Aspekt der Auswertung ist dabei in diesem Fall wie sich die Fehler der Zylinderachse auf die Kreisdetektion auswirken und fortpflanzen.

Analog zur Auswertung der Kreisdetektion bei perfekter Zylinderachse werden alle detektierten Radii in einem Histogramm 5.5 aufgezeigt. Da die Fehler der berechneten Achsen gering sind, kann visuell im Histogramm kein Unterschied zwischen der Darstellung der Radii bei perfekter Achse (siehe Abbildung 5.1) und detektierter Achse festgestellt werden. Ebenfalls fällt kaum auf, dass der erste Balken für die Kreise mit Radius 2,5mm minimal kürzer ist, denn anstatt der gesuchten 4000 Zylinder mit Radius 2,5mm, wurden nur 3999 detektiert. Dies ist, wie zu Beginn bereits erwähnt, auf die geringe Anzahl an Mantelflächenpunkte zurückzuführen. Aufgrund der fehlerbehafteten Achse und somit dem Fehler in den projizierten Punkten, sowie einer möglichen schlechten Detektierung des Kreises durch die RHT, ist es deutlich schwieriger Zylinder mit wenig Mantelflächenpunkten korrekt zu detektieren. Ein weiterer Unterschied ist in dem Histogramm jedoch nicht erkennbar.

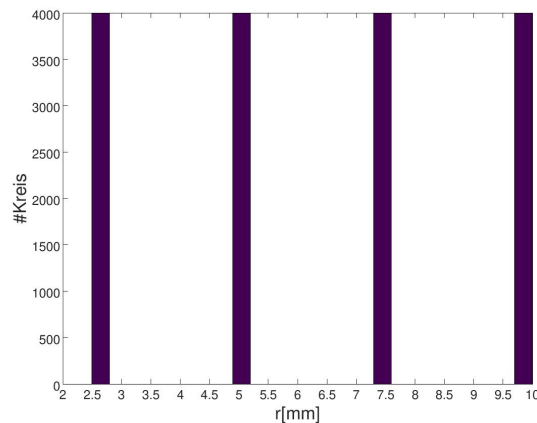


Abbildung 5.5: Darstellung der gefundenen Radii

Aus diesem Grund werden als nächstes die Fehler der verschiedenen Radii einzeln betrachtet. Hierfür werden die Fehler in Histogrammen in Abbildung 5.6 dargestellt. In ihnen kann bereits ein Unterschied zwischen der Kreisdetektion bei perfekter und bei fehlerbehafteter Achse erkannt werden. Anstatt der in Abbildung 5.2 sehr nahe beieinanderliegenden vier Fehlerbalken mit jeweils einer Höhe von 1000, sind nun unterschiedlich viele Fehler aufgezeigt. Dies hängt damit zusammen, dass nun nicht nur zwischen Radius und Höhe des Zylinders unterschieden wird, sondern auch wie groß der eigentliche Fehlerwinkel zwischen der absoluten sowie detektierten Achse ist. Je größer der Fehler der Achse, desto größer ist auch die Ungenauigkeit der Detektion.

Radius [mm]	Mittelwert Radius [mm]	Mittlerer absoluter Fehler [mm]	Standardabweichung [mm]
2,5	2,4997	7,3633e-04	7,6322e-04
5,0	4,9995	5,7031e-04	4,7695e-04
7,5	7,4990	9,5295e-04	1,9202e-04
10,0	9,9984	1,5972e-03	1,5551e-04

Tabelle 5.4: Mittelwert, absoluter Fehler und Standardabweichung der vier Radii

Dahingegen ist der Fehler des Mittelwerts im Vergleich zu der Detektion mit optimaler Zylinderachse kleiner geworden (siehe Tabelle 5.4). Dies liegt jedoch nur daran, dass aufgrund der fehlerbehafteten Achse sowohl positive als auch negative Radiusfehler auftreten. Aus diesem Grund wird in den folgenden Auswertungen immer der mittlere absolute Fehler der Messung betrachtet. An diesem kann gut erkannt werden, dass der durchschnittliche Fehler des Radius deutlich größer als bei vorgegebener perfekter Zylinderachse ist. Trotzdem ist der detektierte Radius sehr nahe an dem eigentlichen absoluten Wert.

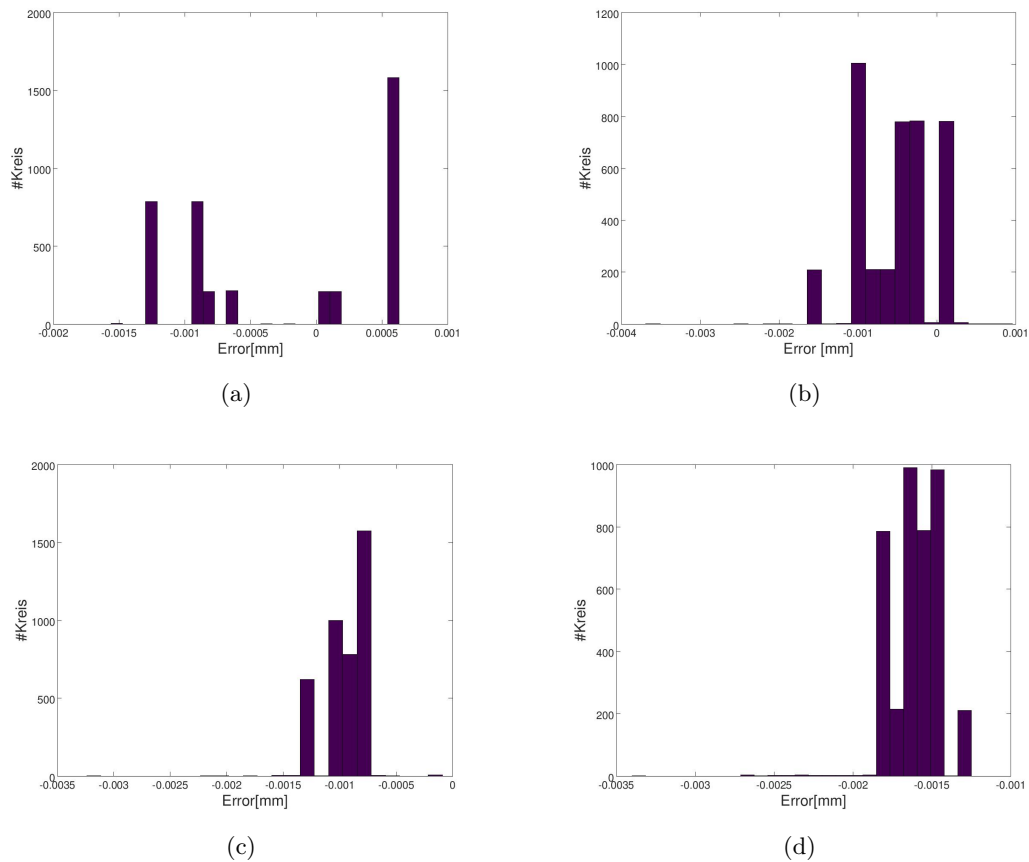


Abbildung 5.6: Fehler der einzelnen Radii: (a) Fehler bei Kreisen mit Radius 2,5; (b) Fehler bei Kreisen mit Radius 5,0; (c) Fehler bei Kreisen mit Radius 7,5; (d) Fehler bei Kreisen mit Radius 10.

Anhand der Standardabweichung kann zudem ein weiterer Trend beobachtet werden. Die Standardabweichung und somit die Streuweite des Radius ist größer je kleiner der Radius ist. Dies unterstützt die Behauptung, dass die Detektion von Kreisen mit wenigen fehlerbehafteten projizierten Mantelpunkten deutlich schwieriger ist und somit auch größere Streuungen zeigt. Sind nur sehr wenige Punkte vorhanden, ist auch eine Verbesserung durch den Least-Square Ansatz deutlich schwerer, da sich Fehler aufgrund der geringen Anzahl an Messpunkten weniger gegenseitig aufheben können.

Insgesamt kann jedoch festgehalten werden, dass die Radii trotz fehlerbehafteter Achse erfolgreich detektiert werden.

Fehler bei der Positionsbestimmung

Nach der Auswertung der Radiusbestimmung wird nun ein genaueres Augenmerk auf die detektierte Position des Zylinders im dreidimensionalen Raum geworfen. Analog zur Auswertung der Position der detektierten Zylinder für eine perfekte Achse, wird hierfür der dreidimensionale Punkt auf der Zylinderachse entlang dieser projiziert und anschließend mit einem absoluten Sollwert verglichen.

Auf eine visuelle Darstellung aller projizierten Punkte in einem Plot wird verzichtet, da die Ungenauigkeiten bei der Bestimmung der Kreismittelpunkte sehr gering sind. Stattdessen werden mithilfe der absoluten Koordinaten die Fehler der Kreismittelpunkte berechnet und in Abbildung 5.7 dargestellt. In der Abbildung links sind alle 15999 Fehler der Kreismittelpunkte auf der Ebene in den projizierten Koordinaten $[u, v]$ aufgetragen. Aufgrund des Least-Square Ansatzes und der simulierten Punktwolke werden für die projizierten Positionen nur etwa 90 einzigartige Fehlerpaare gefunden. In den Abbildungen rechts sind die Fehlerdistanzen jeder berechneten Position in einem Histogramm zusammengefasst. Der maximale Distanzfehler beträgt dabei 0,00474mm, der durchschnittliche Fehler 0,000889mm.

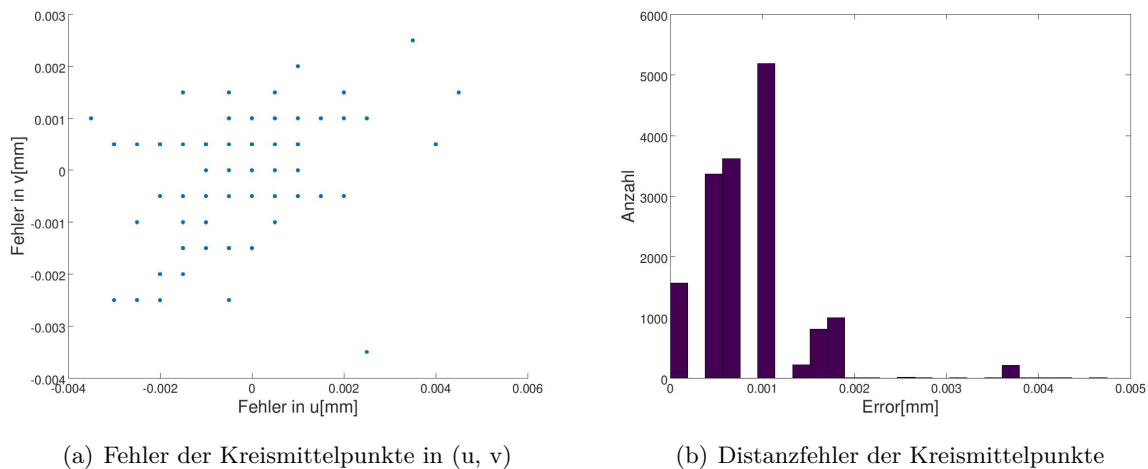


Abbildung 5.7: Fehler der Kreismittelpunktbestimmung

Die Abbildungen 5.7 zeigt bereits sehr gut, dass die berechneten Positionen, zwar nicht exakt den absoluten Werten entsprechen, der Fehler jedoch sehr gering ist. In der Tabelle 5.5 werden noch einmal die Mittelwerte der Kreismittelpunkte, der mittlere Distanzfehler sowie die Standardabweichung dargestellt, dabei wird dieselbe Zylindernummerierung wie bei den Versuch zuvor verwendet (dargestellt in Abbildung 5.3). Anhand dieser ist zu erkennen, dass ein Fehler der projizierten Positionen vorliegt, dieser jedoch sehr klein und stets in einem annehmbaren Bereich liegt.

Auswertung der Radius- und Positionsbestimmung

Insgesamt kann eindeutig festgehalten werden, dass die Zylinderdetektion auf simulierten Daten erfolgreich ist. Die Fehler der Kreisdetektion sind sehr gering und haben eine sehr geringe Streuweite. Aus 16000 Zylindern wurden insgesamt 15999 Zylinder ohne große Fehler detektiert. Die Funktionalität der Kreisdetektion ist somit verifiziert. Aus diesem Grund werden in den folgenden Experimenten Zylinderdetektionen sowohl auf Simulationsdaten, als auch Realdaten ausgewertet.

Zylinder [Radius, Höhe]	Optimale Position [u,v][mm]	Mittelwert [u,v][mm]	Mittlerer Distanzfehler [mm]	Standard- abweichung [mm]
Zyl. 1 [10 mm, 25mm]	[-45,0 -45,0]	[-44,99979 -45,00000]	0,01517	[0,00043 0,00010]
Zyl. 2 [10 mm, 20mm]	[-15,0 -45,0]	[-14,99989 -45,00029]	0,01620	[0,00022 0,00041]
Zyl. 3 [10 mm, 15mm]	[15,0 -45,0]	[15,00021 -44,99990]	0,01625	[0,00041 0,00021]
Zyl. 4 [10 mm, 10mm]	[45,0 -45,0]	[44,99971 -45,00000]	0,01584	[0,00041 0,00002]
Zyl. 5 [7,5 mm, 25mm]	[-45,0 -15,0]	[-45,00088 -15,00059]	0,05733	[0,00123 0,00082]
Zyl. 6 [7,5 mm, 20mm]	[-15,0 -15,0]	[-14,99980 -15,00029]	0,01892	[0,00043 0,00042]
Zyl. 7 [7,5 mm, 15mm]	[15,0 -15,0]	[14,99942 -15,00029]	0,03536	[0,00082 0,00042]
Zyl. 8 [7,5 mm, 10mm]	[45,0 -15,0]	[44,99971 -14,99990]	0,01622	[0,00041 0,00020]
Zyl. 9 [5,0 mm, 25mm]	[-45,0 15,0]	[-45,00040 14,99952]	0,03736	[0,00028 0,00101]
Zyl. 10 [5,0 mm, 20mm]	[-15,0 15,0]	[-14,99950 14,99931]	0,03171	[0,00005 0,00061]
Zyl. 11 [5,0 mm, 15mm]	[15,0 15,0]	[15,00069 15,00030]	0,03340	[0,00063 0,00043]
Zyl. 12 [5,0 mm, 10mm]	[45,0 15,0]	[44,99971 15,00029]	0,02230	[0,00041 0,00041]
Zyl. 13 [2,5 mm, 25mm]	[-45,0 45,0]	[-45,00050 45,00019]	0,02131	[0,00002 0,00061]
Zyl. 14 [2,5 mm, 20mm]	[-15,0 45,0]	[-14,99900 45,00040]	0,03408	[0,00002 0,00020]
Zyl. 15 [2,5 mm, 15mm]	[15,0 45,0]	[15,00113 45,00098]	0,05274	[0,00122 0,00102]
Zyl. 16 [2,5 mm, 10mm]	[45,0 45,0]	[44,99971 45,00050]	0,01984	[0,00041 0,00004]

Tabelle 5.5: Mittlerer Fehler und Standardabweichung der Zylinder Positionen

5.2 Experiment und Auswertung der Zylinderdetektion

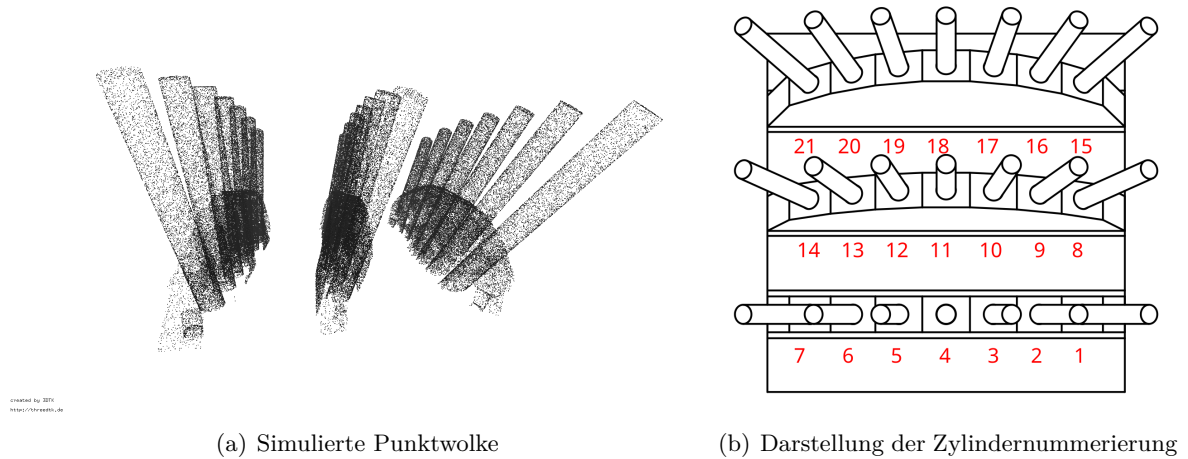


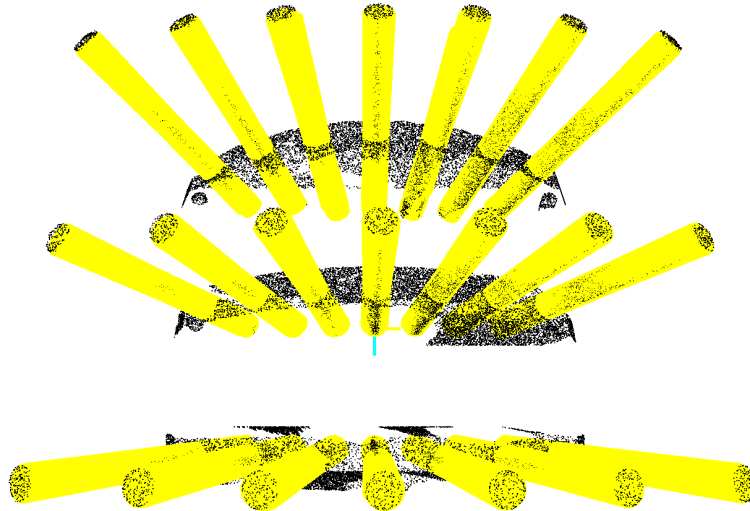
Abbildung 5.8: Darstellung des Modells 2 zur Auswertung der Zylinderdetektion

Im vorigen Abschnitt 5.1.2 wurde die Grundfunktionalität der Kreisdetektion bei einer detektierten Zylinderachse getestet. Aufgrund des kombinierten Schrittes wurde somit die Zylinderdetektion im einfachsten Fall einer einzigen vorhandenen Achse getestet. In den folgenden Abschnitten soll validiert werden, dass Zylinder auch mit verschiedenen Achsen detektierbar sind, selbst wenn einige Ebenenpunkte vorhanden sind.

Hierfür wurde ein Modell mit insgesamt 21 verschiedenen Zylinderachsen und einem Radius von 4mm modelliert. Dabei sind die 21 Zylinder gleichmäßig verteilt auf drei Reihen angebracht. Jede Zylinderreihe besitzt einen einheitlichen Winkel von 90° , 75° oder 60° zur Modellebene. Wobei die Winkeldifferenz zwischen zwei Zylindernachbarn auf einer Reihe stets 10° ist. Im Anhang Abbildung 6.2 wird das Modell mit seinen Winkeln erneut dargestellt.

Für die Zylinderdetektion wurde das Modell gesampelt und die großen Ebenen, wie etwa die Modellplatte, aus der Punktwolke entfernt. Um die Robustheit der Detektion testen zu können, wurden hierbei jedoch einige Ebenenpunkte in der Punktwolke gelassen. In der linken Abbildung von 5.8 wird die simulierte Punktwolke dargestellt, aufgrund der seitlichen Ansicht ist es gut möglich die drei Zylinderreihen zu erkennen. In der rechten werden mit rot den einzelnen Zylindern Zahlen zugeordnet. Diese Zahlen werden im in den Folgenden Abschnitten für die Auswertung verwendet. Dabei stehen die Zylinder 1-7 auf Reihe 1, die Zylinder von 8-14 auf Reihe 2 und die restlichen auf Reihe 3. In den folgenden Abschnitten soll auf dieser Punktwolke die Zylinderdetektion getestet und ausgewertet werden.

5.2.1 Generierung der Testdaten und Aufteilung der detektierten Zylinder



created by 30TK
<http://threadk.de>

Abbildung 5.9: Darstellung der gefundenen Zylinder (gelb) in simulierter Punktvolke

Wie auch bei der Validierung der Kreisdetektion wird zuerst mithilfe des 3D-Modells eine gesampelte 3D-Punktwolke erzeugt. Nach der Entfernung der Grundebenen, besteht die Punktwolke aus etwas mehr als 160 000 Punkten. Wie bereits zuvor wird die Zylinderdetektion aufgrund ihrer Randomisierung auf insgesamt 1000 Detektionen ausgewertet. Dies ermöglicht eine bessere Aussage über die Funktionalität sowie Robustheit der Zylinderdetektion treffen zu können. Daher folgt für den Testdatensatz, dass jeder Zylinder 1000 Mal gefunden werden muss, wodurch insgesamt 21000 Zylinder in allen Durchläufen detektiert werden müssen.

Insgesamt wurden 21580 Zylinder gefunden, davon sind 685 Falschdetektionen und 20895 korrekt detektierte Zylinder, somit sind 105 Fehldetektionen vorhanden. In Tabelle 5.6 sind alle gefundenen Zylinder nach Anzahl ihrer Detektionen so wie ihrem mittleren Radius und ihrer mittleren Zylinderachse aufgeschlüsselt. Sollte eine Achse im Hough-Raum zwei Maxima besitzen wurde eines der Maxima zur Mittelwertberechnung auf die andere Seite des Hough-Raumes gespiegelt. Die aufsummierte Anzahl aller gefundenen Zylinder entspricht dabei der Anzahl der 20895 gefundenen Detektionen.

Aus den 20895 Detektierten, sowie den eigentlich vorhandenen 21000 Zylinder, können die 105 fehlenden berechnet werden. Der Grund für diese liegt größtenteils an der Zylinderachsen-detektion, deren Hough-Raum aufgrund der Ebenenpunkte beeinflusst wird. Dies führt dazu, dass anstatt der 21 eigentlichen Maxima, häufig zwischen 63 und 89 mögliche Zylinderachsen der Kreisdetektion übergeben werden. Die meisten dieser Achsen stammen von Maxima generiert durch Zylinder, jedoch gibt es auch ein paar die durch die Ebenenpunkte generiert werden und

keiner Zylinderachse entsprechen. Diese können zu Falschdetektionen führen. Betrachtet man die Anzahl der einzelnen Detektionen in 5.6 so wurde Zylinder 13 insgesamt 23 Mal nicht gefunden und auch einige andere Zylinder wie 1, 7, 9 und 14 wurden mindestens 10 Mal nicht detektiert. Der Median und gleichzeitig die maximale Anzahl an detektierten Zylindern in einem Durchlauf beträgt 21. Die minimale Anzahl an richtig detektierten Zylindern in einem Durchlauf beträgt 19. In Abbildung 5.9 wird zur besseren Visualisierung jeder 25. detektierte Zylinder dargestellt.

Zylinder [Reihe]	Anzahl	Mittlerer Radius [mm]	Mittlere Zylinderachse
Zyl. 1 [R1]	984	4,017412	[-0,000005 0,867644 0,497098]
Zyl. 2 [R1]	1000	4,019130	[0,000922 0,939864 0,341480]
Zyl. 3 [R1]	999	4,021614	[0,001319 0,984756 0,173802]
Zyl. 4 [R1]	1000	4,021055	[0,000038 0,999972 0,006095]
Zyl. 5 [R1]	1000	4,023453	[-0,001655 -0,984742 0,173892]
Zyl. 6 [R1]	1000	4,018585	[-0,001355 -0,939866 0,341473]
Zyl. 7 [R1]	989	4,017419	[-0,001086 -0,867577 0,497213]
Zyl. 8 [R2]	995	4,012951	[-0,223956 0,836439 0,500157]
Zyl. 9 [R2]	990	4,017824	[-0,243564 0,907487 0,342173]
Zyl. 10 [R2]	992	4,018680	[-0,255035 0,951598 0,171397]
Zyl. 11 [R2]	1000	4,024651	[0,258363 -0,966008 0,007524]
Zyl. 12 [R2]	999	4,022630	[0,253674 -0,951957 0,171428]
Zyl. 13 [R2]	977	4,014294	[0,243128 -0,907709 0,341907]
Zyl. 14 [R2]	987	4,008037	[0,224230 -0,836934 0,499206]
Zyl. 15 [R3]	994	4,012634	[-0,432120 0,74952 0,501454]
Zyl. 16 [R3]	999	4,015868	[-0,470141 0,814086 0,340869]
Zyl. 17 [R3]	1000	4,023035	[-0,491832 0,853793 0,170533]
Zyl. 18 [R3]	1000	4,014919	[0,498872 -0,866663 0,003528]
Zyl. 19 [R3]	999	4,014306	[0,491193 -0,854039 0,171127]
Zyl. 20 [R3]	999	4,012021	[0,469666 -0,814438 0,340685]
Zyl. 21 [R3]	992	4,006345	[0,432675 -0,749493 0,501005]

Tabelle 5.6: Detektionsanzahl der einzelnen Zylinder

Neben den Fehldetektionen ist besonders die große Anzahl an Falschdetektionen auffällig. In 1000 Durchläufen wurden 685 falsche Zylinder detektiert. Diese falschen Zylinder teilen sich jedoch stets die gleichen Eigenschaften. So wurden zwar für die 685 Zylinder 99 Achsen detektiert, diese besitzen jedoch alle einen z-Wert um 0.997. Gleichzeitig weisen alle Falschdetektionen einen sehr kleinen Radius im Bereich von $[0,66699; 1,6654]$ auf. Somit gibt es zwei Möglichkeiten wie der Algorithmus verbessert werden kann. Zum einen können weitere Bedingungen in der Kreisdetektion wie etwa ein minimaler Radius helfen kleinere Zylinder

in Ausreißern zu vermeiden, wobei dies vor allem bei einer falsch detektierten Zylinderachse wichtig ist, die dazu führt das jeder projizierte Punkt ein Ausreißer ist. Zum anderen kann eine bessere Auswertung des Gaußschen Hough-Raumes dazu führen, dass deutlich weniger falsche Zylinderachsen detektiert werden. So können bei der Auswertung etwa weitere Bedingungen an die Nachbarschaft des Maximums gestellt werden, sodass nur punktförmige Maxima detektiert werden.

In den folgen Abschnitten werden die erwähnten Schwierigkeiten erneut aufgegriffen und anhand der einzelnen Schritte genauer ausgewertet. Hierbei wird ein besonderes Augenmerk auf den Grund der Schwierigkeiten sowie auf mögliche Verbesserungen gelegt.

5.2.2 Auswertung Zylinderachsendetektion auf rauschfreier Punktwolke

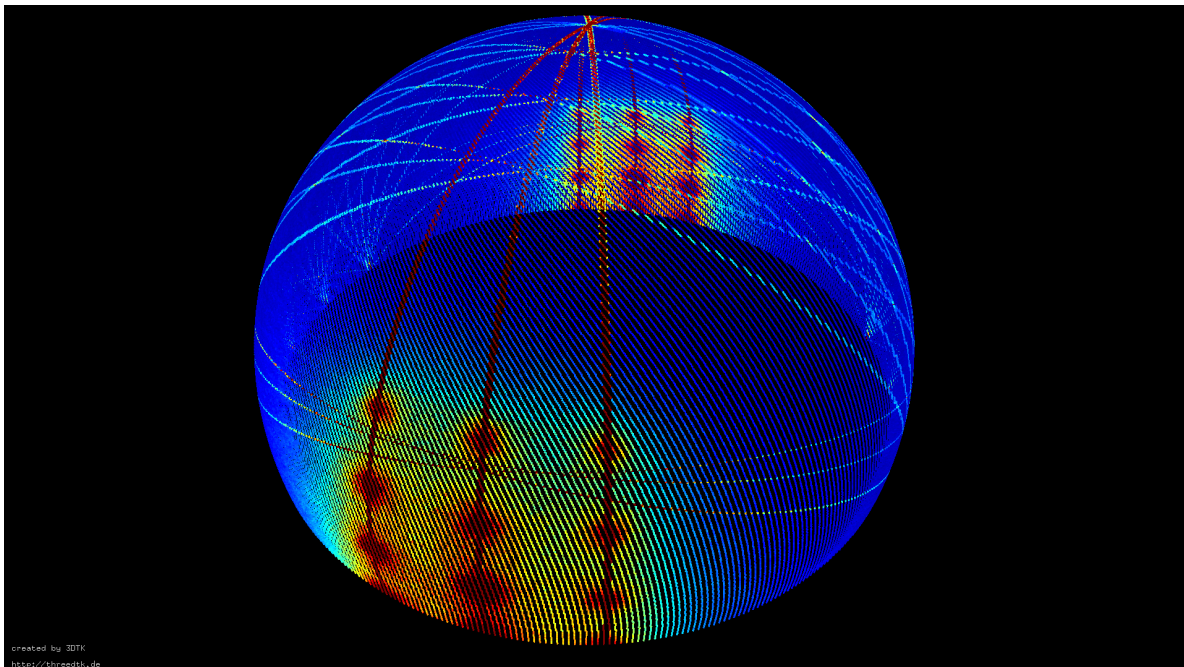


Abbildung 5.10: Darstellung des Akkumulators zur Detektion der Zylinderachsen, wobei blau nicht inkrementiert und rot maximal akkumulierte Zellen beschreibt

Der erste Schritt einer erfolgreichen Zylinderdetektion ist die Detektion von starken Hypothesen für die Zylinderachsen. Das besondere im Falle des für die Zylinderdetektion verwendeten Modells, ist dass stets 7 Zylinder in einer Reihe stehen, die alle denselben Winkel zur Bodenplatte besitzen und nur auf dieser Reihe unterschiedlich zueinander geneigt sind. Betrachtet man den dadurch entstehenden Akkumulator, dargestellt in Abbildung 5.10, so können klar die drei Reihen mit den jeweils sieben Zylindern erkannt werden, wobei 3 der Maxima am Rande der Halbsphäre liegen, weswegen insgesamt $21+3$ Maxima detektiert werden.

Neben den punktförmigen Maxima sind jedoch auch kreisförmige Strukturen zu erkennen. Diese entstehen durch die verbliebenen Ebenenpunkte der Halterung. Insgesamt teilen sich die Punkte auf 6 verschiedene Ebenen auf, wobei stets zwei Ebenen den gleichen Normalenvektor besitzen. So entstehen durch die Ebenenpunkte drei Kreise auf dem Akkumulator. Aus diesem Grund müssen große Ebenen vor der Zylinderdetektion aus der Punktwolke entfernt werden. Kleinere Überreste sind jedoch kein großes Problem, vor allem wenn die Maxima der Zylinder deutlich größer sind, als etwa die der Ebenen. In diesem Fall sind jedoch so viele Ebenenpunkte vorhanden, dass diese Einfluss auf die Zylinderachsendetektion nehmen. Dies führt zu breiteren Maxima der Zylinderachsen im Hough-Raum und gleichzeitig zu falschen Hypothesen die um $[0\ 0\ 1]$ liegen. Letzteres kann schließlich zur Detektion von kleinen Zylindern in Ausreißerhäufungen führen.

Eine Möglichkeit falsche Zylinderachsendetektionen zu umgehen, kann durch eine Erweiterung der zur Auswertung verwendeten Filterfensterfunktion möglich sein. Hierfür müssen nicht nur wie im Moment die nächsten Nachbarn, sondern auch das weitere Umfeld in Betracht gezogen werden, um den Aufbau eines Maximums zu analysieren. Weist das Maximum eine linienförmige Kreisstruktur auf muss dieses entfernt werden, ist es dagegen punktförmig, so handelt es sich wahrscheinlich um die Hypothese einer Zylinderachse. Eine solche Erweiterung des Filters würde die Robustheit der Zylinderachsendetektion stark steigern. Gleichzeitig würde es die anschließend nötigen Kreisdetektionen minimieren, da die gefundene Achsenanzahl sinkt. In dem speziellen Fall dieser Auswertung würden zudem die falsch detektierten Zylinder wegfallen, da sich deren Achse durch den Schnittpunkt der drei Kreise im Hough-Raum beschreiben lässt.

Zylinder	Anzahl Achsen	Mittlerer Zylinderachse Soll Zylinderachse	Mittlerer absoluter Fehler[°]	Standardabweichung
Zyl. 1 [R1]	21	[-0,000005 0,867644 0,497098] [0,000000 0,866025 0,500000]	0,008753	[0,001704 0,004579 0,007978]
Zyl. 2 [R1]	23	[0,000922 0,939864 0,341480] [0,000000 0,939693 0,342020]	0,006115	[0,001880 0,002225 0,006114]
Zyl. 3 [R1]	17	[0,001319 0,984756 0,173802] [0,000000 0,984808 0,173648]	0,005957	[0,001886 0,001140 0,006420]
Zyl. 4 [R1]	20	[0,000038 0,999972 0,006095] [0,000000 1,000000 0,000000]	0,006613	[0,001796 0,000022 0,003976]
Zyl. 5 [R1]	20	[-0,001655 -0,984742 0,173892] [-0,000000 -0,984808 0,173648]	0,005793	[0,001602 0,001103 0,006208]
Zyl. 6 [R1]	23	[-0,001355 -0,939866 0,341473] [-0,000000 -0,939693 0,342020]	0,006146	[0,001558 0,002259 0,006211]
Zyl. 7 [R1]	20	[-0,001086 -0,867577 0,497213] [-0,000000 -0,866025 0,500000]	0,008828	[0,001419 0,004606 0,008020]
Zyl. 8 [R2]	21	[-0,223956 0,836439 0,500157] [-0,224144 0,836516 0,500000]	0,005919	[0,001964 0,003524 0,006385]
Zyl. 9 [R2]	23	[-0,243564 0,907487 0,342173] [-0,243210 0,907673 0,342020]	0,006244	[0,001760 0,002501 0,007252]

Fortsetzung auf nächster Seite

Tabelle 5.7 – Fortsetzung von zu vorgehender Seite

Zylinder	Anzahl Achsen	Mittlerer Zylinderachse Soll Zylinderachse	Mittlerer absoluter Fehler[°]	Standardabweichung
Zyl. 10 [R2]	25	[-0,255035 0,951598 0,171397] [-0,254887 0,951251 0,173648] [0,258363 -0,966008 0,007524]	0,005781	[0,001722 0,001024 0,006054]
Zyl. 11 [R2]	20	[0,258819 -0,965926 -0,000000]	0,007952	[0,002052 0,000563 0,003977]
Zyl. 12 [R2]	22	[0,253674 -0,951957 0,171428] [0,254887 -0,951251 0,173648]	0,005832	[0,001321 0,001181 0,006016]
Zyl. 13 [R2]	23	[0,243128 -0,907709 0,341907] [0,243210 -0,907673 0,342020]	0,005856	[0,001929 0,002440 0,006648]
Zyl. 14 [R2]	20	[0,224230 -0,836934 0,499206] [0,224144 -0,836516 0,500000]	0,005863	[0,001842 0,003482 0,006299]
Zyl. 15 [R3]	16	[-0,432120 0,749520 0,501454] [-0,433013 0,750000 0,500000]	0,005041	[0,001541 0,002719 0,005160]
Zyl. 16 [R3]	21	[-0,470141 0,814086 0,340869] [-0,469846 0,813798 0,342020]	0,005442	[0,001073 0,002368 0,005796]
Zyl. 17 [R3]	26	[-0,491832 0,853793 0,170533] [-0,492404 0,852869 0,173648]	0,006836	[0,001813 0,001161 0,007289]
Zyl. 18 [R3]	20	[0,498872 -0,866663 0,003528] [0,500000 -0,866025 -0,000000]	0,004209	[0,001586 0,000913 0,002419]
Zyl. 19 [R3]	29	[0,491193 -0,854039 0,171127] [0,492404 -0,852869 0,173648]	0,007186	[0,001430 0,001404 0,007723]
Zyl. 20 [R3]	19	[0,469666 -0,814438 0,340685] [0,469846 -0,813798 0,342020]	0,005324	[0,001972 0,001880 0,005702]
Zyl. 21 [R3]	22	[0,432675 -0,749493 0,501005] [0,433013 -0,750000 0,500000]	0,005553	[0,001873 0,003158 0,005889]

Tabelle 5.7: Mittlere Zylinderachsen, Fehler und Standardabweichungen

In der Tabelle 5.7 sind die einzelnen mittleren Achsen der Zylinder sowie ihre Standardabweichung aufgetragen. Im Falle der doppelten Maxima wurden die Zylinderachsen auf eine Seite des Hough-Raumes gespiegelt, sodass der Mittelwert sinnvoll ist und die Standardabweichung eine Aussage über die Streuweite treffen kann. Im Gegensatz zur Auswertung der Kreisdetektion werden deutlich mehr Achsen (etwa 20 pro Zylinder) über alle Durchläufe gefunden. Diese Anzahl steigt jedoch noch weiter betrachtet man die später folgende Auswertung auf realen Daten, wo meist über 50 teilweise sogar über 100 Achsen für einen Zylinder möglich sind (vgl. Tabelle 5.11). Da im Vergleich zu dem ersten Modell deutlich mehr Ebenenpunkte vorhanden sind, welche die detektierten Maxima beeinflussen, sind 20 mögliche Achsen bei einer Hough-Raum Größe von 636473 Zellen eine weiterhin geringe Anzahl. An den Standardabweichungen ist zudem gut erkennbar, dass die Streuweite der Achsen sehr gering ist.

Somit kann für die Zylinderachsendetektion trotz störender Ebenenpunkte festgehalten werden, dass die Zylinderachsen erfolgreich mit nur einer geringen Streuweite detektiert wurden. Gleichzeitig wurden aufgrund der Ebenenpunkte jedoch auch falsche Achsen detektiert, die anschließend zu Falschdetektionen führen können. Eine Möglichkeit diese Schwäche zu verbessern kann durch eine erweiterte Auswertefunktion des Akkumulators erfolgen. Ohne diese Erweiterung muss für eine bessere Hypothese der Zylinderachsen, insbesondere bei komplexeren und größeren Datensätzen, eine sehr genaue Ebenendetektion im Vorfeld berechnet und die Punkte auf den detektierten Ebenen gelöscht werden.

5.2.3 Auswertung Kreisdetektion auf rauschfreier Punktvolke

Neben einer erweiterten Auswertung des Akkumulators, ist es auch möglich die falschen Zylinder durch weitere Bedingungen in der Kreisdetektion, etwa durch das Vorgeben eines minimalen Radius, zu minimieren.

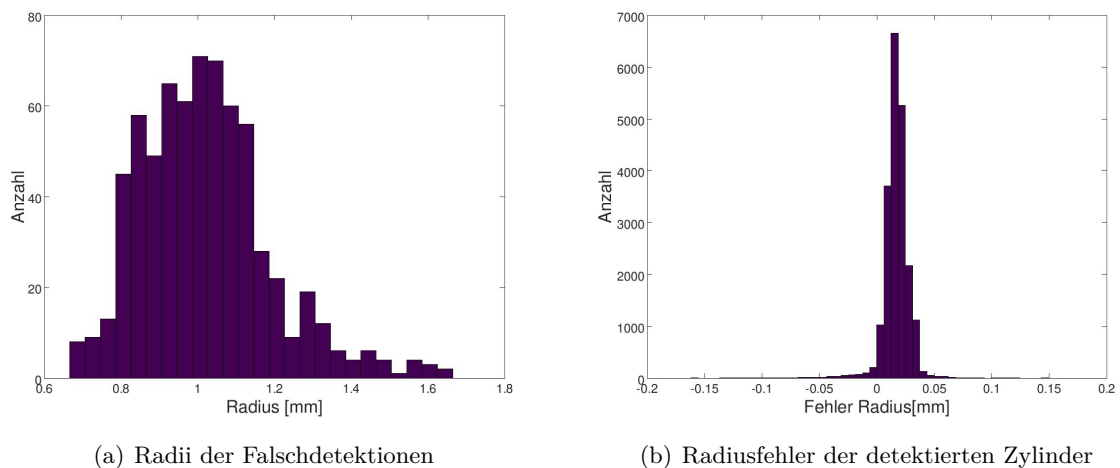


Abbildung 5.11: Detektierte Radii der Falschdetektionen (links) und Fehler der korrekt detektierten Kreise (rechts)

Durch Betrachtung der Radiusverteilung der falschen Zylinder dargestellt in dem linken Abbild von 5.11 wird ersichtlich warum dies möglich ist. Die meisten falsch detektierten Zylinder besitzen einen sehr kleinen Radius bei etwa 1,0mm, während die korrekt detektierten Zylinder nur einen sehr kleinen Fehlerbereich aufweisen (vergleiche rechte Abbildung 5.11) und somit alle bei etwa 4.0mm liegen. Eine Anpassung der Validierungsparameter kann dieses Problem minimieren. Erhöht man etwa die maximale Anzahl an positiven Sektoren eines Kreises, die für eine erfolgreiche Validierung nötig sind, so kann man Falschdetektionen minimieren. Gleichzeitig erhöht eine solche Anpassung der Validierungsparameter jedoch auch die Anzahl der Fehldetektionen. Eine andere Möglichkeit bietet sich in der Realität zudem oft, da der größte falsch detektierte Kreis einen Radius von 1,6654mm und der Kleinste einen von 0,66699mm besitzt,

somit kann etwa auch durch die einfache Einführung eines minimalen Radius von zum Beispiel 2,0 mm, die Anzahl an Falschdetektionen minimiert werden.

Ein solche Bedingung verhindert jedoch nicht nur die Detektion von sehr kleinen, falschen Zylindern, sondern ermöglicht zudem die Laufzeit der Kreisdetektion zu verbessern, gerade wenn zudem ebenfalls ein maximaler Radius etwa von 8,0mm gesetzt wird. Diese zwei Bedingungen ermöglichen es, Kreise deren Radii sicher falsch sind, nicht in den wachsenden Akkumulator einzufügen, sondern direkt nach der Berechnung zu löschen. So muss der Akkumulator nicht unnötig groß aufgespannt werden und die Kreisdetektion kann schneller auf mögliche Einträge testen. Im Falle dieser Masterarbeit wird jedoch stets der allgemein gültige Fall ohne weitere Bedingungen betrachtet, da das Hinzufügen von Bedingungen stets die Zeit und das Ergebnis verbessert.

Zylinder	Radiusbereich [mm]	Mittlerer Radius[mm]	Mittlerer absoluter Fehler [mm]	Standardabweichung Radius [mm]
Zyl. 1 [R1]	[3,873570, 4,042860]	4,017412	0,021955	0,018160
Zyl. 2 [R1]	[3,920040, 4,052510]	4,019130	0,020352	0,008557
Zyl. 3 [R1]	[3,867790, 4,048450]	4,021614	0,022870	0,011747
Zyl. 4 [R1]	[3,926180, 4,041820]	4,021055	0,022128	0,007909
Zyl. 5 [R1]	[3,880710, 4,101550]	4,023453	0,024644	0,010546
Zyl. 6 [R1]	[3,938140, 4,077900]	4,018585	0,020057	0,010217
Zyl. 7 [R1]	[3,843660, 4,048110]	4,017419	0,021130	0,016531
Zyl. 8 [R2]	[3,898780, 4,053980]	4,012951	0,014993	0,011855
Zyl. 9 [R2]	[3,838110, 4,086050]	4,017824	0,019991	0,014621
Zyl. 10 [R2]	[3,905640, 4,058190]	4,018680	0,019494	0,008811
Zyl. 11 [R2]	[3,931470, 4,045540]	4,024651	0,025484	0,009415
Zyl. 12 [R2]	[3,896700, 4,060400]	4,022630	0,023936	0,011261
Zyl. 13 [R2]	[3,937590, 4,117260]	4,014294	0,015215	0,011965
Zyl. 14 [R2]	[3,893810, 4,024210]	4,008037	0,010173	0,010058
Zyl. 15 [R3]	[3,953360, 4,034800]	4,012634	0,013547	0,006042
Zyl. 16 [R3]	[3,906410, 4,051200]	4,015868	0,016917	0,009431
Zyl. 17 [R3]	[3,878660, 4,149400]	4,023035	0,024821	0,015317
Zyl. 18 [R3]	[3,976850, 4,051830]	4,014919	0,015092	0,004304
Zyl. 19 [R3]	[3,903860, 4,124240]	4,014306	0,015507	0,012861
Zyl. 20 [R3]	[3,918530, 4,032940]	4,012021	0,013412	0,008599
Zyl. 21 [R3]	[3,901680, 4,033120]	4,006345	0,007770	0,006264

Tabelle 5.8: Mittlerer Radius, absoluter Fehler und Standardabweichung der detektierten Radii

In Tabelle 5.8 werden die Radii aller korrekt detektierten Zylinder aufgetragen. Dabei kann anhand der zweiten Spalte gut erkannt werden, dass die Radiusbereiche sehr klein sind. Der größte gefundene Radius liegt bei 4,149400mm (Zyl. 17) und der kleinste bei 3,838110mm (Zyl. 9), somit ist der maximale Bereich über alle Zylinder gerade einmal 0,31129mm groß. Auch die Standardabweichungen zeigen gut, dass die Streuweite der Radii gering ist. So besitzt etwa Zylinder 1 mit 0,018160mm die größte Standardabweichung. Somit spiegelt die Tabelle die Erkenntnisse wieder, die aus der visuellen Auswertung von Abbildung 5.11 bereits gewonnen wurden. Die Fehler und Streuweite der detektierten Radii sind sehr gering, somit liegen sie stets um den Wert 4mm, während alle Falschdetektionen einen deutlich kleineren Radiusbereich aufweisen.

Zylinder	Mittlerer Achsenpunkt [mm]	Abstandsbereich zur Achse [mm]	Mittlere Distanz [mm]
Zyl. 1 [R1]	[42,502098 41,822152 65,257740]	[0,000507 0,199956]	0,015829
Zyl. 2 [R1]	[42,499282 44,087813 41,970432]	[0,002480 0,363393]	0,010053
Zyl. 3 [R1]	[42,497363 45,511804 20,584587]	[0,002861 0,240293]	0,008630
Zyl. 4 [R1]	[42,501955 46,229488 0,001379]	[0,002023 0,123379]	0,007813
Zyl. 5 [R1]	[42,499416 45,773216 -20,631825]	[0,001331 0,337462]	0,007752
Zyl. 6 [R1]	[42,501714 43,972568 -41,927370]	[0,001814 0,169967]	0,009328
Zyl. 7 [R1]	[42,500599 41,874985 -65,289221]	[0,001063 0,224524]	0,012072
Zyl. 8 [R2]	[-9,433853 38,929871 65,778028]	[0,001950 0,181119]	0,009469
Zyl. 9 [R2]	[-10,073417 41,327897 42,373780]	[0,002593 0,267479]	0,014194
Zyl. 10 [R2]	[-13,282702 53,281860 22,686828]	[0,000809 0,150547]	0,023826
Zyl. 11 [R2]	[-10,641500 43,457008 -0,003666]	[0,002319 0,204778]	0,013622
Zyl. 12 [R2]	[-10,476494 42,821427 -20,797641]	[0,002299 0,241416]	0,011043
Zyl. 13 [R2]	[-10,059727 41,293272 -42,362794]	[0,003254 0,240355]	0,027521
Zyl. 14 [R2]	[-9,908514 40,730444 -66,855537]	[0,000307 0,165650]	0,022076
Zyl. 15 [R3]	[-60,923384 36,906518 65,271119]	[0,004304 0,139589]	0,014409
Zyl. 16 [R3]	[-62,406250 39,489644 42,237565]	[0,001984 0,415306]	0,012354
Zyl. 17 [R3]	[-63,747417 41,798975 20,920701]	[0,003202 0,300633]	0,023859
Zyl. 18 [R3]	[-63,407932 41,181783 0,003090]	[0,002213 0,076114]	0,013272
Zyl. 19 [R3]	[-63,276835 40,955831 -20,760079]	[0,003521 0,207553]	0,028102
Zyl. 20 [R3]	[-63,031954 40,547248 -42,690749]	[0,006646 0,171834]	0,021634
Zyl. 21 [R3]	[-61,929947 38,647564 -66,435912]	[0,002536 0,153352]	0,020544

Tabelle 5.9: Fehlerdistanzen der Zylinderachsenpunkte

Neben dem Radius soll die Kreisdetektion ebenfalls einen Punkt auf der Zylinderachse bestimmen, um die 3D-Position des Zylinders im Raum festzulegen. Im Gegensatz zur Auswertung der Kreisdetektion zu Beginn, oder der späteren Auswertung der halben Zylinder, ist ein Vergleich mit absoluten Werten in diesem Fall nur sehr schwer möglich, da alle Zylinder eine einzigartige Achse besitzen. Somit besitzt jeder Zylinder eine eigene Projektionsebene, wobei der Ursprung abhängig vom berechneten Koordinatensystem ist.

Um trotzdem die Punkte auf der Zylinderachse auszuwerten, wird aus den End- beziehungsweise Anfangspunkten der Zylinder ein Zylindermittelpunkt berechnet. Anschließend wird die Distanz zur Zylinderachse, definiert durch den berechneten Zylindermittelpunkt und der absoluten Zylinderachse kalkuliert. Da die Auswertung abhängig ist von dem berechneten mittleren Zylindermittelpunkt, kann keine Aussage über einen systematischen Fehler getroffen werden. Da jedoch weder im Abschnitt 5.1.2, noch im Abschnitt 5.3 ein systematischer Fehler in der Kreismittelpunktbestimmung auftritt, wird angenommen, dass dieser wenn er existiert, deutlich kleiner als der zufällige Fehler ist.

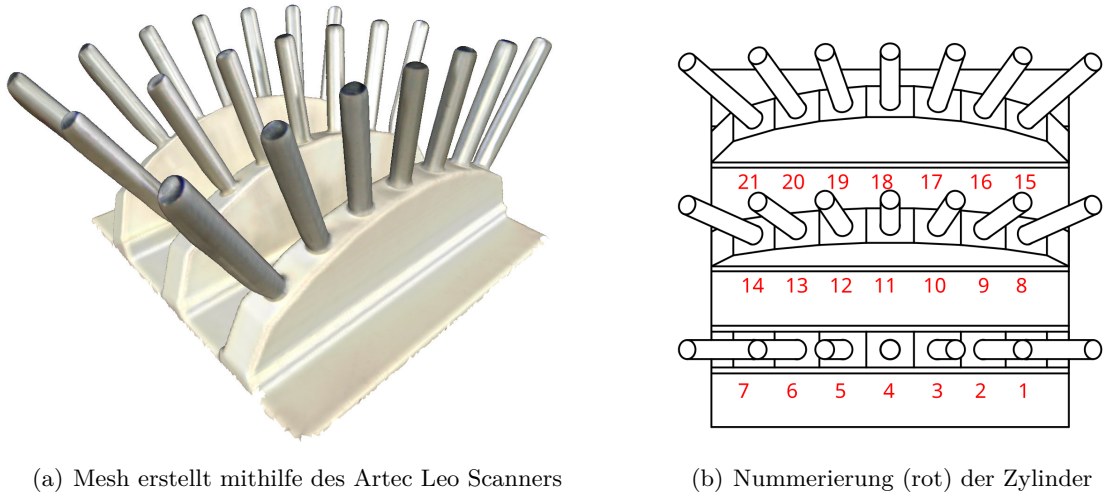
In Tabelle 5.9 sind die Referenzpunkte der mittleren Achsenpunkte angegeben. Diese und die Zylinderachse beschreiben die Gerade auf welcher alle Punkte liegen müssen. Neben dem mittleren Achsenpunkt, wird in der Tabelle auch die minimale und maximale, sowie die mittlere Distanz zu dieser Achse angegeben. Dabei kann gut erkannt werden, dass die Streuweite der einzelnen Punkte wie auch bereits die Streuweite der Radii minimal ist. Aufgrund der fehlenden absoluten Referenzmittelpunkte, gibt die Tabelle jedoch keine Aussage über die absolute Abweichung, wodurch systematische Fehler nicht erkannt werden können. Der Abstandsbereich sowie die mittlere Distanz geben jedoch einen groben Referenzwert für den zufälligen Fehler an.

Anhand des geringen Abstandsbereiches zur Achse und den Radii mit sehr kleiner Streuweite, kann festgehalten werden, dass die Kreisdetektion trotz störender Ebenenpunkte erfolgreich ist. Die Auswertung zeigt jedoch auch auf, dass falsche Zylinderachsen zu falschen Detektionen in Ausreißern führen kann, die meist einen sehr kleinen Radius besitzen. Diese können durch strenger gewählte Validierungsparameter, etwa eine höher gesetzte Anzahl an positiven Sektoren oder durch das Hinzufügen von Bedingungen minimiert werden.

5.2.4 Bewertung Zylinderdetektion auf rauschfreier Punktwolke

Zusammengefasst kann festgehalten werden, dass der Algorithmus Zylinder mit verschiedenen Achsen erfolgreich detektiert werden kann. Ähnliche Zylinderachsen können in einem passend gewählten Akkumulator gut dargestellt werden. Klar anzumerken ist jedoch auch, dass eine große Menge an Ebenenpunkten eindeutig zu Problemen führt. Selbst in diesem Beispiel, wo etwa die große Bodenplatte komplett entfernt wurde und nur kleinere Artefakte der Halterung vorhanden sind, konnten Auswirkungen auf den Hough-Raum festgestellt werden. So mussten in dem Falle des simulierten Datensets etwa deutlich mehr Zylinderachsen für die Kreisdetektion betrachtet werden. Zudem können einige dieser Achsen falsch sein, was bereits bei Simulationsdaten zu falschen Kreisen und somit Falschdetektionen führen kann. Bei einem realen Datensatz ist somit die Detektion, wenn Ebenenpunkte vorhanden sind, besonders schwierig.

5.2.5 Generierung realer Scan und Kategorisierung der detektierten Zylinder



(a) Mesh erstellt mithilfe des Artec Leo Scanners

(b) Nummerierung (rot) der Zylinder

Abbildung 5.12: Mesh und Zylindernummerierung des 2. Modells

Um die Zylinderdetektion auch auf realen Daten testen zu können, wurde das 3D-Modell (dargestellt in Abbildung 6.2) bis auf die Zylinder 3D-gedruckt. Anstelle der modellierten Zylinder wurden in den 3D-Druck Alustäbe mit einem Durchmesser von ungefähr 8,05 mm (gemessen mit einem Messschieber) eingesetzt. Für die Auswertung wurde dieselbe Zylinder-nummerierung wie für die Auswertung der Simulationsdaten, die Nummerierung wird noch einmal in der rechten Abbildung 5.12 dargestellt.

Mithilfe eines Artec Leo Scanners wurde das gedruckte Modell vollständig gescannt. Die daraus entstandenen einzelnen Teilaufnahmen des Objektes, wurden anschließend zu einer Aufnahme wieder zusammengefügt. Der Artec Leo besitzt nach Hersteller Angaben eine 3D-Punktgenauigkeit von bis zu 0,01mm und eine 3D-Auflösung von bis zu 0,05mm. Die Punktwolken werden jedoch einer Nachbearbeitung unterzogen welche helfen soll, Lücken im Modell mithilfe von Interpolation zu schließen und ungenaue Scanabschnitte zu bereinigen. [5]

Die Nachbearbeitung besitzt jedoch den großen Nachteil, dass die harten Kanten zwischen den Zylindermantelflächen und den Zylinderkreisen als Rundung interpretiert werden, wodurch die durch den Hersteller angegebene Genauigkeiten nicht erreicht werden kann. In Abbildung 5.13 werden mit rot zwei kritische Stellen in der Punktwolke umrandet. Zum einen die Spitze des Zylinders, die aufgrund der Nachbearbeitung, eine Rundung aufweist, zum anderen wird das Zylinderende markiert, welches einen deutlich zu kleinen Radius besitzt. Da diese Nachbearbeitung bereits die Zylinderdetektion erschwert, wurden die meisten Ebenenpunkte auf der Halterung, sowie die Modellbodenplatte entfernt. Ohne die Ebenenpunkte besteht der rechtshändige Scan aus 424711 Punkten. Wie bereits zuvor wurde nicht nur eine Detektion durchgeführt, sondern insgesamt 1000 Durchläufe zur Auswertung genutzt. Der Mittelwert der Ausführzeit für einen Durchlauf liegt dabei bei 43,7s (verwendete CPU: AMD Ryzen 5 2600X).

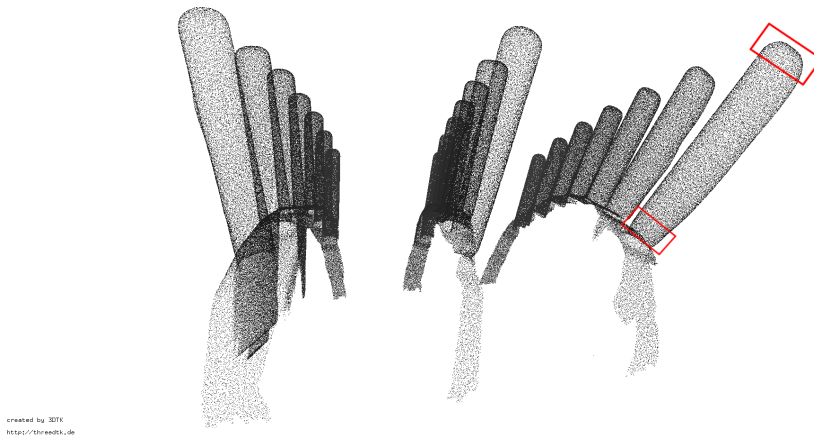


Abbildung 5.13: Reale Punktwolke des Modells 2, rot: Beispiele für kritische Stellen entstanden durch die Nachbearbeitung der Artec Leo Software

In 1000 Durchläufen wurden statt der erwarteten 21000 Zylinder insgesamt 21125 Zylinder gefunden. Davon sind 20858 Zylinder richtig detektiert, somit fehlen 142 Detektionen. Zylinder 19 mit 895 und Zylinder 20 mit 977 erfolgreichen Detektionen wurden am seltensten gefunden, während die restlichen Zylinder alle zwischen 996 und 1000 erfolgreiche Detektionen aufweisen können. In Abbildung 5.14 werden die mittleren Zylinder der 1000 Detektionen dargestellt. Man kann gut erkennen, dass aufgrund der Interpolation, die oberen kugelförmigen Zylinderenden nicht detektiert werden. Auch einige Zylinder die am Rande einer Reihe stehen, weisen nahe der Befestigungen einen offensichtlich zu kleinen Radius auf und können somit nicht vollständig erfasst werden.

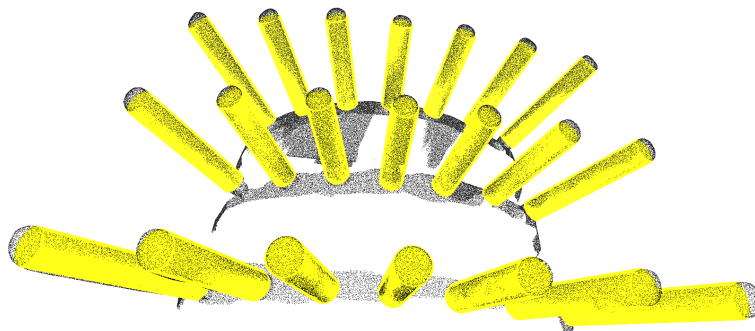


Abbildung 5.14: Zylinder (gelb) berechnet aus den Mittelwerten der detektierten Zylinder

267 berechnete Zylinder sind Falschdetektionen, wobei diese Anzahl im Vergleich zur Simulation aufgrund der kleineren Menge an Ebenenpunkten gesunken ist. Die Falschdetektionen besitzen wie auch bereits bei den Simulationsdaten kleine Radii. Insgesamt besitzen 224, also etwa 83,9% aller Falschdetektionen einen Radius kleiner als 2,5mm. Mögliche Lösungen für dieses Problem wurden bereits in der Auswertung der simulierten Daten angesprochen.

Aus diesem Grund werden in den folgenden zwei Abschnitten noch einmal die Fehler der Zylinderachsendetektion sowie der Kreissuche genauer betrachtet. Dabei wird noch einmal auf die Unterschiede zwischen den Simulations- sowie Realdaten eingegangen.

Zylinder	Anzahl	Mittelwert Radius [mm]	Mittlerer Achse
Zyl. 1 [R1]	998	4,385837	[0,338382 0,373748 0,863479]
Zyl. 2 [R1]	1000	4,212868	[0,230884 0,259546 0,937531]
Zyl. 3 [R1]	1000	4,091209	[0,122332 0,123496 0,984657]
Zyl. 4 [R1]	1000	3,961372	[0,014676 -0,008713 0,999824]
Zyl. 5 [R1]	1000	3,976495	[-0,117097 -0,151404 0,981366]
Zyl. 6 [R1]	1000	4,057594	[-0,226783 -0,265343 0,936940]
Zyl. 7 [R1]	1000	4,131697	[-0,333609 -0,376526 0,864110]
Zyl. 8 [R2]	999	4,285496	[0,514952 0,221345 0,828044]
Zyl. 9 [R2]	1000	4,132170	[0,418994 0,082546 0,904182]
Zyl. 10 [R2]	1000	4,059775	[0,318332 -0,045989 0,946818]
Zyl. 11 [R2]	999	4,039723	[0,205805 -0,182902 0,961279]
Zyl. 12 [R2]	1000	4,057945	[0,088732 -0,310165 0,946475]
Zyl. 13 [R2]	999	4,107037	[-0,028213 -0,425533 0,904458]
Zyl. 14 [R2]	1000	4,120827	[-0,154003 -0,534709 0,830790]
Zyl. 15 [R3]	998	4,391449	[0,665148 0,077799 0,742590]
Zyl. 16 [R3]	1000	4,353024	[0,584651 -0,057298 0,809168]
Zyl. 17 [R3]	997	4,192846	[0,490671 -0,205716 0,846610]
Zyl. 18 [R3]	996	4,139120	[0,380471 -0,339377 0,860142]
Zyl. 19 [R3]	895	4,146973	[0,259587 -0,468589 0,844249]
Zyl. 20 [R3]	977	4,200681	[0,126788 -0,577034 0,806622]
Zyl. 21 [R3]	1000	4,233473	[-0,000824 -0,670143 0,742200]

Tabelle 5.10: Detektierte Zylinder und ihre mittleren Radii beziehungsweise Zylinderachsen

5.2.6 Auswertung Zylinderachsendetektion auf realen Daten

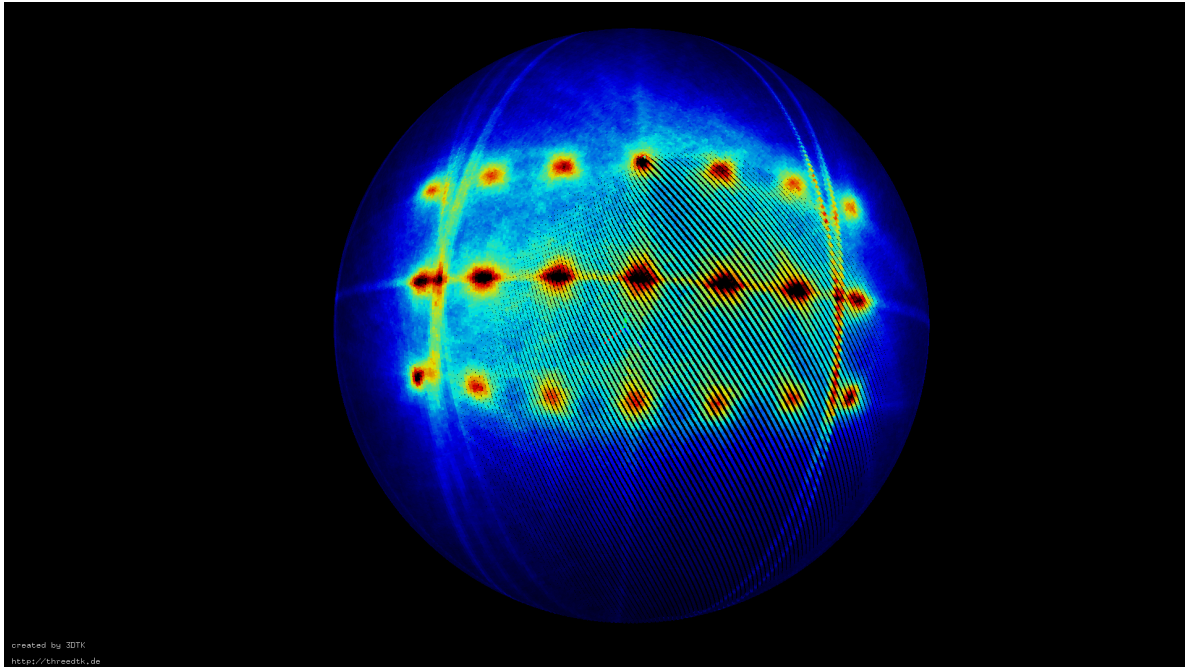


Abbildung 5.15: Darstellung des Akkumulators zur Detektion der Zylinderachsen, wobei blau nicht inkrementiert und rot maximal viele akkumulierte Zellen beschreibt

Wie auch schon zu Beginn der Zylinderachsenauswertung auf den Simulationsdaten wird zu Beginn der Akkumulator, dargestellt in Abbildung 5.15, betrachtet. In diesem Fall wurde das vom Laserscanner erzeugte, rechtshändische Koordinatensystem direkt verwendet, ohne es als linkshändisches Koordinatensystem zu exportieren. Aus diesem Grund befinden sich die Maxima alle auf der oberen Seite der Akkumulator Halbkugel. Aufgrund der deutlich geringeren Ebenenpunkte, sind die Maxima Kreise der Ebenen im Vergleich zu dem Simulationsdatensets deutlich weniger stark ausgeprägt.

In der Tabelle 5.11 sind die mittleren Zylinderachsen sowie ihre Standardabweichung und die Anzahl ihrer verschiedenen Zylinderachsen dargestellt. Im Vergleich zu den simulierten Daten besitzen die meisten detektierten realen Zylinder eine deutlich größere Menge an gefundenen Achsen. Insgesamt wurden für die verschiedenen Zylinder zwischen 26 und 206 verschiedene Achsen gefunden. Betrachtet man die Standardabweichung der einzelnen Werte, so kann eindeutig festgehalten werden, dass die Streuweite der detektierten Achsen für die Realdaten im Vergleich zu den Simulationsdaten deutlich zugenommen hat. Jedoch ist die Streuweite immer noch in einem kleinen Bereich. Ein direkter Vergleich mit absoluten Werten ist bei realen Datensets nicht möglich, da nur relative Referenzen in einem Modell bekannt sind.

Zylinder	Anzahl Achsen	Mittlerer Zylinderachse	Standard Abweichung
Zyl. 1 [R1]	65	[0,338382 0,373748 0,863479]	[0,009994 0,009671 0,004576]
Zyl. 2 [R1]	72	[0,230884 0,259546 0,937531]	[0,014009 0,012168 0,004527]
Zyl. 3 [R1]	68	[0,122332 0,123496 0,984657]	[0,009738 0,011709 0,001748]
Zyl. 4 [R1]	26	[0,014676 -0,008713 0,999824]	[0,005959 0,005114 0,000066]
Zyl. 5 [R1]	66	[-0,117097 -0,151404 0,981366]	[0,010726 0,012786 0,002629]
Zyl. 6 [R1]	75	[-0,226783 -0,265343 0,936940]	[0,012130 0,012012 0,003783]
Zyl. 7 [R1]	64	[-0,333609 -0,376526 0,864110]	[0,011376 0,009990 0,004198]
Zyl. 8 [R2]	159	[0,514952 0,221345 0,828044]	[0,006906 0,009553 0,005933]
Zyl. 9 [R2]	92	[0,418994 0,082546 0,904182]	[0,005380 0,006971 0,002871]
Zyl. 10 [R2]	102	[0,318332 -0,045989 0,946818]	[0,005562 0,007131 0,001781]
Zyl. 11 [R2]	129	[0,205805 -0,182902 0,961279]	[0,007664 0,008546 0,001702]
Zyl. 12 [R2]	125	[0,088732 -0,310165 0,946475]	[0,007410 0,007085 0,002060]
Zyl. 13 [R2]	105	[-0,028213 -0,425533 0,904458]	[0,006704 0,005485 0,002697]
Zyl. 14 [R2]	135	[-0,154003 -0,534709 0,830790]	[0,008337 0,007392 0,005728]
Zyl. 15 [R3]	111	[0,665148 0,077799 0,742590]	[0,004503 0,007016 0,004062]
Zyl. 16 [R3]	135	[0,584651 -0,057298 0,809168]	[0,006733 0,008901 0,004819]
Zyl. 17 [R3]	177	[0,490671 -0,205716 0,846610]	[0,007571 0,009809 0,004645]
Zyl. 18 [R3]	205	[0,380471 -0,339377 0,860142]	[0,009543 0,009606 0,006113]
Zyl. 19 [R3]	204	[0,259587 -0,468589 0,844249]	[0,012762 0,008977 0,006298]
Zyl. 20 [R3]	193	[0,126788 -0,577034 0,806622]	[0,013882 0,009460 0,005982]
Zyl. 21 [R3]	73	[-0,000824 -0,670143 0,742200]	[0,004011 0,004168 0,003760]

Tabelle 5.11: Mittlere Zylinderachse und Standardabweichung der einzelnen Zylinder

Aus diesem Grund müssen die Winkel zwischen den einzelnen Achsen betrachtet werden. Jeder Zylinder besitzt mindestens zwei Referenzzyylinder, die sich auf gleicher Position in den zwei anderen Reihen, verschoben um einen Winkel von 15° beziehungsweise 30° , befinden. Zudem ist bekannt, dass der Zylinder und seine Nachbarn auf einer Reihe eine Winkeldifferenz von 10° besitzen.

Die mittlere Winkeldifferenz zwischen den Nachbarn einer Reihe und zwischen den Referenzzyindern auf den anderen Reihen ist in den drei Tabellen 5.12, 5.13 und 5.14 für jede Reihe aufgetragen. Neben der Winkeldifferenz kann aus den Tabellen auch der relative absolute mittlere Fehler abgelesen werden. Wird der Fehler der relativen Winkeldifferenz zwischen den Zylindern einer Reihe betrachtet, so ist der maximale Fehler gegeben durch $1,7016^\circ$ (Reihe 1), der minimale $0,42775^\circ$ (Reihe 3), und der Durchschnitt $0,70012^\circ$.

Um die Winkeldifferenz zwischen den Zylindern der verschiedenen Reihen zu betrachten, wird die mittlere Reihe 2 mit Reihe 1 und Reihe 3 verglichen. Dies hat den Vorteil, dass die Winkeldifferenz zwischen den Zylindern und ihren Referenzzylindern auf der anderen Reihe stets 15° entspricht. Der maximale Fehler zwischen den Referenzzylindern von Reihe 2 zu den anderen Reihen beträgt $2,10785^\circ$, der minimale $0,26532^\circ$ und der Durchschnitt $0,98230$.

Wird die Verteilung der Fehler der einzelnen Zylinder in den Tabellen 5.12, 5.13 und 5.14 betrachtet, fällt auf, dass die Zylinder am Reihenrand, einen deutlich größeren Winkeldifferenzfehler zu ihrem Nachbarn auf der Reihe, so wie auch zu den zwei Referenzzylindern auf den anderen Reihen, aufweisen. Der Durchschnitt der Fehlerwinkeldifferenz zwischen den Zylinderreihen ohne die zwei Randzylinder beträgt $0,69712^\circ$, wobei der größte Fehler gerade einmal $1,33778^\circ$ beträgt.

Zylinder [auf R1]	Nachbar auf R2 Winkel[°]; Fehler[°]	Nachbar auf R3 Winkel[°]; Fehler[°]	Nachbar R1 Winkel[°]; Fehler[°]
Zyl. 1	13,604144; 1,417434	26,481514; 3,518486	NaN; NaN
Zyl. 2	15,293039; 0,882916	28,807846; 1,235808	9,898373; 0,487555
Zyl. 3	14,998413; 0,538410	29,838211; 0,649033	10,479885; 0,688597
Zyl. 4	15,040488; 0,289654	29,570354; 0,571482	9,791816; 0,549321
Zyl. 5	14,948919; 0,265315	29,701710; 0,461109	11,694049; 1,701605
Zyl. 6	14,635565; 0,792643	28,149161; 1,864442	9,037731; 1,020495
Zyl. 7	13,817539; 1,186344	26,625785; 3,374215	9,759507; 0,614175

Tabelle 5.12: Relative Winkel der Zylinder auf Reihe 1

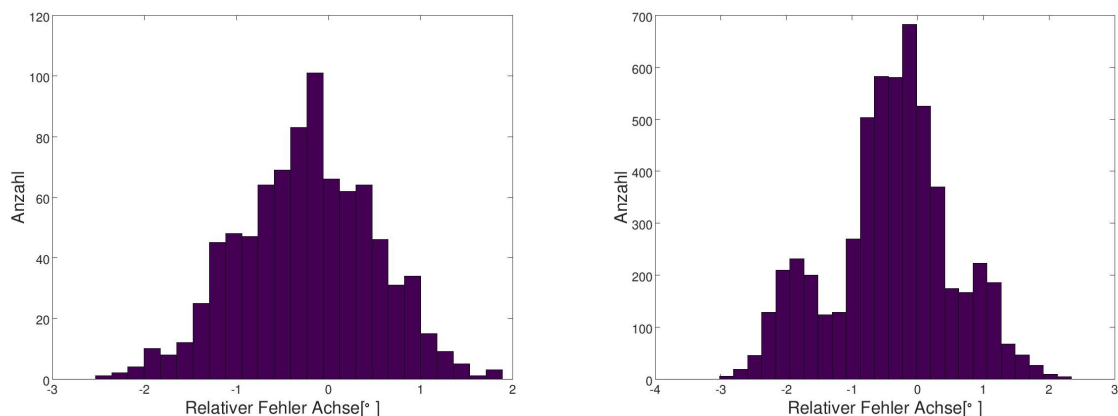
Zylinder [auf R2]	Nachbar auf R1 Winkel[°]; Fehler[°]	Nachbar auf R3 Winkel[°]; Fehler[°]	Nachbar R2 1 Winkel[°]; Fehler[°]
Zyl. 8	13,604144; 1,417434	12,930680; 2,069320	NaN; NaN
Zyl. 9	15,293039; 0,882916	13,664055; 1,337778	10,615550; 0,824108
Zyl. 10	14,998413; 0,538410	14,809296; 0,471629	9,471513; 0,628227
Zyl. 11	15,040488; 0,289654	14,559609; 0,633119	10,412181; 0,589210
Zyl. 12	14,948919; 0,265315	14,903793; 0,427439	9,975459; 0,589201
Zyl. 13	14,635565; 0,792643	13,669088; 1,332292	9,715666; 0,570653
Zyl. 14	13,817539; 1,186344	12,892150; 2,107850	10,728588; 0,813749

Tabelle 5.13: Relative Winkel der Zylinder auf Reihe 2

Zylinder [auf R3]	Nachbar auf R2 Winkel[°]; Fehler[°]	Nachbar auf R1 Winkel[°]; Fehler[°]	Nachbar R3 1 Winkel[°]; Fehler[°]
Zyl. 15	12,930680; 2,069320	26,481514; 3,518486	NaN; NaN
Zyl. 16	13,664055; 1,337778	28,807846; 1,235808	10,014320; 0,427746
Zyl. 17	14,809296; 0,471629	29,838211; 0,649033	10,202482; 0,593827
Zyl. 18	14,559609; 0,633119	29,570354; 0,571482	9,947050; 0,545626
Zyl. 19	14,903793; 0,427439	29,701710; 0,461109	10,250782; 0,569886
Zyl. 20	13,669088; 1,332292	28,149161; 1,864442	10,191009; 0,748745
Zyl. 21	12,892150; 2,107850	26,625785; 3,374215	9,777007; 0,639433

Tabelle 5.14: Relative Winkel der Zylinder auf Reihe 3

Dies ist ein deutlicher Unterschied zu dem zuvor berechneten gesamten Fehlerdurchschnitt von 0,98230 und dem maximalen Fehler von 2,10785°. Der Grund hierfür ist mitunter, dass im Hough-Raum in der Nähe der Maxima der Randzylinderachsen Kreise von den verbliebenen Ebenenpunkten liegen. Diese können die Maxima beeinflussen und eine genaue Detektion deutlich erschweren. Dies kann ebenfalls in der Abbildung des Akkumulators 5.15 gut erkannt werden.



(a) Relativer Fehler der Zylinderachsen auf Reihe 1

(b) Relativer Fehler zwischen Reihe 1 und Reihe 2

Abbildung 5.16: Darstellung der relativen Fehlerwinkel der Zylinderachsen

Die Ergebnisse der Tabellen zeigen sich auch in den Abbildungen 5.16 wieder. In der linken Abbildung werden die Fehler der relativen Winkel zwischen den Zylindern der 1. Reihe dargestellt. Die Verteilung lässt in diesem Fall klar erkennen dass die relativen Fehler einen Peak um 0 besitzen. In der Abbildung rechts hingegen werden die Fehler der Winkeldifferenz zwischen Reihe 2 und Reihe 1 sowie Reihe 2 und 3 dargestellt. Auffällig sind dabei die kleinen Peaks neben dem großen Maximum bei 0, welche hauptsächlich von den zwei Randzylindern stammen.

Trotz der steigenden Fehler der Randzylinder ist die Zylinderachsendetektion gut genug, um eine Kreisdetektion anhand der Punktwolke und den detektierten Zylinderachsen zu ermöglichen. Nur durch herausfiltern jeglicher Ebenenpunkte, würde die Zylinderachsendetektion noch weiter profitieren. Da jedoch in den meisten realen Fällen Ebenen vorkommen, sollten spätere Implementation eine sehr genaue Ebenendetektion, welche selbst kleine Ebenen entfernen kann besitzen oder eine bessere Auswertung des Hough-Raumes nutzen.

5.2.7 Auswertung Kreisdetektion auf realen Daten

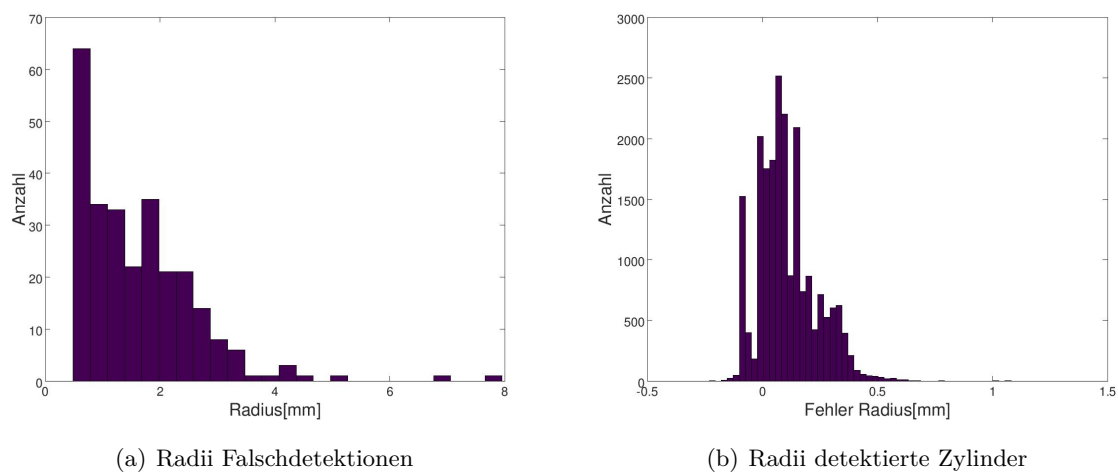


Abbildung 5.17: Vergleich der Radii der Falschdetektionen (links) mit den Radii der korrekt detektierten Zylinder(rechts)

Nicht nur die Streuweite der Zylinderachsen hat durch die realen Daten deutlich zugenommen, auch die berechneten Radii der Kreisdetektion weisen nun einen deutlich größeren Bereich von $[3,8189 \text{ } 5,1310]$ mm auf. Dies ist auch gut erkennbar in der rechten Abbildung von 5.17. Hier werden die Fehler der gefundenen Radii aufgetragen. Im Vergleich zu dem klaren Peak der Simulationsdaten, sind die Fehler der Radii hier deutlich weiter verstreut, gerade im positiven Bereich.

In der Tabelle 5.15 sind die detektierten Radii und ihre absoluten mittleren Fehler nach Zylinder aufgeschlüsselt. Dabei ist klar zu erkennen, dass die mittleren Fehler der einzelnen Radii deutlich größer sind, als etwa bei den Simulationsdaten. Ebenfalls fällt auf, dass sich der Fehler der Zylinderachsendetektion in den detektierten Radii zeigt. So besitzen etwa die meisten Randzylinder im Vergleich zu den restlichen Zylindern einen relativ großen mittleren Fehler. Berechnet man etwa den Durchschnitt der absoluten mittleren Fehler aller Randzylinder so erhält man $0,2097$ mm, während der absolute mittlere Fehler der restlichen Zylinderradii bei $0,0895$ mm liegt.

Anhand dieses Unterschiedes erkennt man somit gut, dass je höher die Anforderung an die Präzision der Zylinderdetektion ist, desto genauer muss die Anforderung an die Zylinderachse sein, da sich Fehler in dieser direkt in der Kreisdetektion widerspiegeln. Trotz Least-Square Ansatz ist es aufgrund der systematischen Fehler, die durch eine falsche Zylinderachse entstehen, nicht möglich diese vollständig zu beheben.

Zylinder	Radiusbereich [mm]	Mittelwert Radius [mm]	mittlerer absoluter Fehler [mm]	Standardabweichung Radius [mm]
Zyl. 1 [R1]	[3,992840 4,727300]	4,385837	0,336017	0,091645
Zyl. 2 [R1]	[4,020240 4,331050]	4,212868	0,162933	0,027448
Zyl. 3 [R1]	[3,947030 4,138270]	4,091209	0,041717	0,016627
Zyl. 4 [R1]	[3,921470 3,975950]	3,961372	0,088628	0,004806
Zyl. 5 [R1]	[3,883010 4,119870]	3,976495	0,073762	0,020466
Zyl. 6 [R1]	[3,928220 4,216220]	4,057594	0,020845	0,027529
Zyl. 7 [R1]	[3,818860 4,251030]	4,131697	0,089126	0,051653
Zyl. 8 [R2]	[4,101310 4,460480]	4,285496	0,235496	0,038874
Zyl. 9 [R2]	[3,968260 4,179010]	4,132170	0,082712	0,015291
Zyl. 10 [R2]	[3,981010 4,089620]	4,059775	0,010301	0,005675
Zyl. 11 [R2]	[3,894860 4,097630]	4,039723	0,011510	0,013063
Zyl. 12 [R2]	[3,953280 4,142590]	4,057945	0,010726	0,013018
Zyl. 13 [R2]	[4,000650 4,151010]	4,107037	0,057738	0,014101
Zyl. 14 [R2]	[3,895330 4,235960]	4,120827	0,072855	0,026004
Zyl. 15 [R3]	[4,189280 5,131040]	4,391449	0,341449	0,078991
Zyl. 16 [R3]	[4,120490 4,548210]	4,353024	0,303024	0,054069
Zyl. 17 [R3]	[4,072990 4,290110]	4,192846	0,142846	0,014754
Zyl. 18 [R3]	[3,955020 4,298450]	4,139120	0,089323	0,016568
Zyl. 19 [R3]	[3,939050 4,299490]	4,146973	0,097226	0,023248
Zyl. 20 [R3]	[4,019010 4,406260]	4,200681	0,150785	0,032162
Zyl. 21 [R3]	[4,016400 4,468200]	4,233473	0,183580	0,044864

Tabelle 5.15: Wiedergabe der detektierten Radii durch ihren Mittelwert, absoluten Fehler und der Standardabweichung

Aufgrund der realen Daten ist ein Vergleich mit absoluten Positionen nicht möglich, auch zu den relativen Positionen der Zylindermittelpunkte gibt es keine exakte Information, da die Höhe der Zylinder und somit der ausgerechnete Mittelpunkt der Zylinder stark schwankt (vergleiche hierfür Werte in Tabelle 5.16). Dies liegt mitunter an der Nachbearbeitung der Scans, die dazu führt, dass etwa aufgrund der Interpolation zwischen Mantelfläche und Halterung sowie Mantelfläche und Kreisfläche der Radius unterschiedlich groß ist. Betrachtet man die mittleren Längen,

so ist der maximale Abstand zwischen den 21 Längen etwa 14,2mm. Der Schwankungsbereich der Zylinderlängen ist somit eindeutig zu groß, als das ein relativer Vergleich der einzelnen Punkte möglich ist.

In Tabelle 5.16 sind deshalb nur die mittleren Punkte auf der Achse, die Länge der Zylinder und die Standardabweichungen der Zylinderachsenpunkte aufgezeigt. Dabei fallen aufgrund der Standardabweichung wieder die Randzylinder auf, da diese eine deutlich größere Standardabweichung als etwa die mittleren Zylinder besitzen. Dies deutet ebenfalls erneut an, was eigentlich durch den Radius bereits gut gezeigt wurde: Je größer der Fehler der Zylinderachse, desto schlechter beziehungsweise ungenauer kann der daraus projizierte Kreis detektiert werden, weswegen auch die Streuung des Mittelpunktes größer wird.

Zylinder	Zylinderhöhe [mm]	Mittelwert Punkt auf Achse [mm]	Standardabweichung der Zylinderachsenpunkte [mm]
Zyl. 1 [R1]	51,089	[40,840 88,016 98,790]	[0,231 0,475 0,225]
Zyl. 2 [R1]	43,046	[24,939 70,038 105,560]	[0,182 0,668 0,171]
Zyl. 3 [R1]	39,419	[9,265 52,074 109,753]	[0,125 0,631 0,118]
Zyl. 4 [R1]	37,505	[-6,522 34,302 110,915]	[0,089 0,488 0,096]
Zyl. 5 [R1]	38,560	[-22,024 16,475 109,700]	[0,358 1,000 0,368]
Zyl. 6 [R1]	41,937	[-37,869 -1,194 106,163]	[0,452 0,921 0,451]
Zyl. 7 [R1]	48,068	[-54,148 -19,387 100,295]	[0,502 0,881 0,496]
Zyl. 8 [R2]	45,007	[83,254 52,340 96,487]	[0,059 0,200 0,059]
Zyl. 9 [R2]	40,385	[67,551 32,558 101,807]	[0,030 0,157 0,030]
Zyl. 10 [R2]	36,860	[52,348 13,818 105,421]	[0,049 0,157 0,050]
Zyl. 11 [R2]	38,161	[36,531 -4,027 105,419]	[0,091 0,237 0,095]
Zyl. 12 [R2]	38,326	[20,696 -21,665 104,688]	[0,112 0,252 0,118]
Zyl. 13 [R2]	40,193	[4,100 -39,088 101,689]	[0,136 0,269 0,140]
Zyl. 14 [R2]	46,969	[-13,243 -55,975 95,543]	[0,259 0,460 0,269]
Zyl. 15 [R3]	48,551	[122,938 15,361 92,132]	[0,020 0,022 0,020]
Zyl. 16 [R3]	41,242	[109,319 -4,872 97,904]	[0,011 0,023 0,011]
Zyl. 17 [R3]	39,279	[94,452 -24,126 100,820]	[0,043 0,094 0,041]
Zyl. 18 [R3]	37,627	[79,115 -42,295 101,846]	[0,104 0,274 0,100]
Zyl. 19 [R3]	38,726	[62,839 -59,469 100,585]	[0,248 0,503 0,239]
Zyl. 20 [R3]	42,684	[45,562 -75,756 96,944]	[0,295 0,482 0,280]
Zyl. 21 [R3]	46,802	[27,258 -91,877 91,835]	[0,204 0,341 0,205]

Tabelle 5.16: Streuweite der Zylinderachsenmittelpunkte

5.2.8 Bewertung Zylinderdetektion auf realen Daten

Mit dem in dieser Masterarbeit vorgestellten Ansatz der Zylinderdetektion ist es möglich Zylinder auch in realen Scans zu finden. Eine Detektion ist ohne Einschränkung der 5 Zylinderparameter möglich, jedoch muss die Punktwolke für eine gute Detektion eine Bedingung erfüllen: Es darf keine große Menge an Ebenenpunkten vorhanden sein. Je mehr Ebenenpunkte in der Punktwolke sind, desto schwieriger ist es den Hough-Raum aufgrund der entstehenden Kreise auszuwerten. Ebenenpunkte können somit nicht nur die Zylinderdetektion erschweren, sondern auch unmöglich gestalten, sollte eine Ebene stärker als die Maxima der einzelnen Zylinder voten. Der in dieser Arbeit implementierte Ansatz ist somit nur bedingt für eine direkte Auswertung der Daten ohne vorherige Ebenendetektion geeignet.

5.3 Experiment und Auswertung der Detektion von halben Zylindern

Im Abschnitt 5.2 wurde die Grundfunktionalität des Algorithmus anhand von Simulations- und Echtdateen eines Modells mit 21 Vollzylindern demonstriert. Ziel der Arbeit ist es jedoch nicht nur ganze, sondern auch Teilzylinder, im schlimmsten Fall halbe Zylinder, detektieren zu können. Dies ist wichtig, da es in realen Umgebungen vorkommen kann, dass ein Teil des Zylinders durch einen Gegenstand zum Beispiel eine Wand unscannbar ist, oder große Schatten eine Seite des Zylinders verdecken. Bei einem Scan wird zudem nie ein kompletter Zylinder gescannt werden, hierfür sind somit stets mehrere notwendig. Aus diesem Grund soll in dieser Auswertung anhand von liegenden Halbzyllindern demonstriert werden, dass es möglich ist, mit dem Algorithmus auch Teilzylinder zu detektieren.

Um die Detektion von Teilzylindern, hier wie im schlechtesten Fall Halbzyllindern, zu testen, wurde ein Modell mit sechs ganzen, sowie 16 halben Zylindern erstellt. In Abbildung 6.3 wird das Modell mit den einzelnen Zylindern sowie Längen dargestellt. Die 16 halben Zylinder sind dabei im Kreis liegend um die sechs stehenden Vollzylinder angeordnet. Alle liegenden Zylinder besitzen eine Länge von 30 mm. Fünf der Halbzyllinder haben einen Radius von 7,5mm, während die restlichen Liegenden sowie Stehenden einen Radius von 5mm besitzen. Der Winkel zwischen den Achsen zweier benachbarter liegender Zylindern mit Radius 5 mm beträgt je 18° , beziehungsweise 30° bei 7,5mm. Zur besseren Verständlichkeit wurden die Winkel ebenfalls noch einmal in 6.3 eingezeichnet.

In dem folgenden Abschnitt wird die Detektionsgenauigkeit und Fehleranfälligkeit des Algorithmus für die Detektion halber Zylinder auf Simulationsdaten bestimmt. Anschließend soll mit Hilfe von Echtdateen die Funktion auf Stabilität getestet werden. Dies ist wichtig da Ausreißer, Rauschen und Schatten die Ergebnisse des Algorithmus stark beeinflussen können, insbesondere im hier vorliegenden Falle halber Zylinder.

5.3.1 Generierung der Testdaten und Aufteilung der detektierten Zylinder

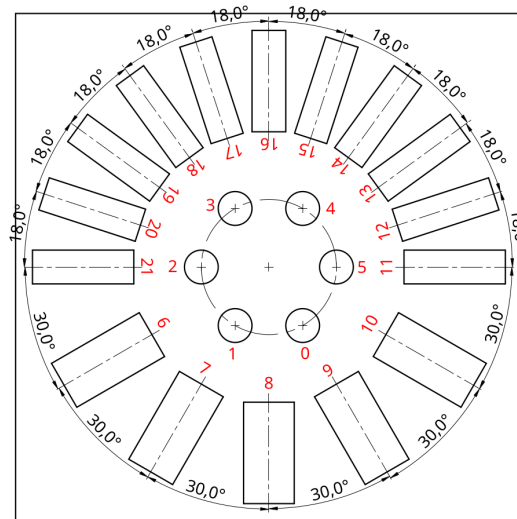
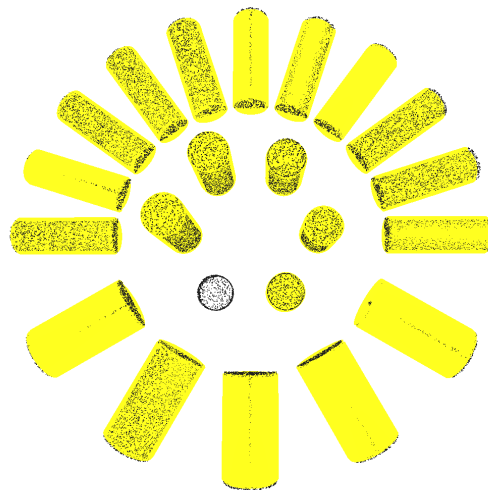


Abbildung 5.18: Zylinder nummerierung (rot) des Modells 3



created by 2019
https://threewiki.de

Abbildung 5.19: Darstellung detektierte Zylinder (gelb) in Punkt看ke

Für die Auswertung der Detektion von halben Zylindern auf Simulationsdaten, wurde das Modell mit etwas mehr als 50000 Punkten gesampelt. Aufgrund des randomisierten Verfahrens wurden wie in den Versuchen zuvor nicht eine einzelne Messung, sondern 1000 Zylinderdetektion auf dieser Punkt看ke durchgeführt. Im Folgenden werden nun zuerst grob die gefundenen Zylinder nach fehlend, falsch und richtig detektiert kategorisiert, anschließend werden im Abschnitt 5.3.2 der erste sequentiellen Schritt der Zylinderdetektion genauer betrachtet. Die Auswertung des Radius und des Punktes auf der Zylinderachse erfolgen im darauf folgenden Abschnitt (5.3.3). Für die Auswertung wird stets die Zylinder nummerierung dargestellt in 5.18 genutzt.

Da das Modell aus 22 Zylindern besteht, wovon 6 Vollzylinder und die restlichen 16 liegende Halbzylinder sind, müssen in 1000 Durchläufen insgesamt 22000 Zylinder detektiert werden. Es wurden jedoch nur 20691 Zylinder gefunden, wobei der beste Durchlauf 21 Zylinder korrekte detektierte Zylinder fand und der schlechteste nur 19. Es gab keine Falschdetektionen, somit sind 20691 von 220000 Zylindern korrekt und 1309 Zylinder nicht detektiert worden. Im Durchschnitt wurden 20,7 Zylinder richtig detektiert.

Zylinder [Radius[mm]; Höhe[mm]]	Ganzer/Halber Zylinder	Anzahl	Mittlerer Radius [mm]	Mittlere Zylinderachse
Zyl. 0 [5; 5]	ganzer Zyl.	0	-	-
Zyl. 1 [5; 10]	ganzer Zyl.	993	4.999	[-0,000785 0,000000 1,000000]
Zyl. 2 [5; 15]	ganzer Zyl.	1000	4.999	[-0,000785 0,000000 1,000000]
Zyl. 3 [5; 20]	ganzer Zyl.	1000	4.999	[-0,000785 0,000000 1,000000]
Zyl. 4 [5; 25]	ganzer Zyl.	1000	4.999	[-0,000785 0,000000 1,000000]
Zyl. 5 [5; 30]	ganzer Zyl.	1000	4.999	[-0,000785 0,000000 1,000000]
Zyl. 6 [7,5; 30]	halber Zyl.	1000	7.498	[0,865826 -0,500338 0,000785]
Zyl. 7 [7,5; 30]	halber Zyl.	1000	7.498	[0,499590 -0,866256 0,000785]
Zyl. 8 [7,5; 30]	halber Zyl.	1000	7.504	[0,000460 -0,999995 0,000785]
Zyl. 9 [7,5; 30]	halber Zyl.	1000	7.500	[-0,500316 -0,865838 0,000785]
Zyl. 10 [7,5; 30]	halber Zyl.	1000	7.497	[-0,865800 -0,500382 0,000785]
Zyl. 11 [5; 30]	halber Zyl.	924	5.002	[0,999997 0,001097 0,000785]
Zyl. 12 [5; 30]	halber Zyl.	1000	4.999	[0,951275 0,308333 0,000785]
Zyl. 13 [5; 30]	halber Zyl.	1000	4.998	[0,809930 0,586518 0,000785]
Zyl. 14 [5; 30]	halber Zyl.	1000	4.998	[0,588398 0,808564 0,000785]
Zyl. 15 [5; 30]	halber Zyl.	1000	4.998	[0,308531 0,951209 0,000785]
Zyl. 16 [5; 30]	halber Zyl.	1000	4.997	[-0,000460 0,999995 0,000785]
Zyl. 17 [5; 30]	halber Zyl.	999	4.957	[-0,309401 0,950926 0,000785]
Zyl. 18 [5; 30]	halber Zyl.	1000	4.953	[-0,589411 0,807827 0,000785]
Zyl. 19 [5; 30]	halber Zyl.	1000	4.998	[-0,809605 0,586967 0,000785]
Zyl. 20 [5; 30]	halber Zyl.	1000	4.998	[-0,951346 0,308112 0,000785]
Zyl. 21 [5; 30]	halber Zyl.	775	5.000	[-1,000000 -0,000097 0,000785]

Tabelle 5.17: Detektionsanzahl der einzelnen Zylinder

Da dieser deutlich kleiner als der Erwartete von 22 ist, werden in Tabelle 5.17 die Anzahl der Detektionen nach den einzelnen Zylinder aufgeschlüsselt. Deutlich erkennbar ist, dass der kleinste Vollzylinder mit Radius 5mm und Höhe 5mm nicht detektiert wird. Auch bei der optischen Betrachtung einer der 1000 Zylinderdetektionen, dargestellt in 5.19, ist deutlich zu erkennen, dass

der kleinste Zylinder nicht markiert wurde. Dies liegt an der geringen Anzahl an Mantelpunkten, die es der Kreisvalidierungsfunktion mit einer Kreisauflösung von 0,1 und mindestens einem Punkt in 25% aller Kreissegmente nicht erlaubt, diesen als Kreis zu validieren.

Eine Möglichkeit diesen kleinsten Zylinder zu detektieren kann die Anpassung der Validierungsparameter der Kreisdetektion ermöglichen. Eine Erhöhung der Schrittweite der Kreisauflösung führt zu weniger Kreissegmenten, wodurch weniger Punkte auf einem Kreis liegen müssen. Für eine Auflösung von 0,2mm wurden bei insgesamt 100 Versuchen jedoch 12 Falschdetektionen beobachtet, die zu jeweils sechs Fehldetektionen geführt haben. Der Grund hierfür ist der kreisförmige Aufbau der stehenden Vollzylinder. Dieser führt etwa bei einer zu niedrigen Kreissegmentauflösung dazu, dass die sechs stehenden Zylinder als ein großer Zylinder mit Radius 40mm detektiert werden. Je nach Anwendungszweck ist es jedoch auch möglich Zylinder mit nur wenigen Mantelpunkten zu detektieren, sollten die Zylinder nicht kreisförmig im Raum angeordnet sein und die Anzahl an Ebenenpunkten sehr klein ist. Da im Folgenden jedoch der Ansatz auch auf Realdaten getestet werden soll, wurde in dieser Auswertung ein robuster Parametersatz für die Kreisvalidierung gewählt, bei der zwar ein Zylinder nicht erkannt wird, jedoch auch keine Falschdetektionen vorliegen.

Trotzdem ist ein Mittelwert von 20,7 immer noch signifikant kleiner als 21. In Tabelle 5.17 wird ersichtlich, dass insgesamt 17 Zylinder 1000 Mal detektiert wurden. Auf die Zylinder 1, 11, 17 und 21 sind die 309 fehlenden Detektionen aufgeteilt, wobei Zylinder 17 nur einmal nicht detektiert wurde und Zylinder 1 7 Fehldetektionen aufweisen kann. Beide Zylinderdetektionen lassen sich etwa durch eine größere Anzahl an Mindestdurchläufen der RHT für die Kreisdetektion weiter verbessern.

Dies wäre bei Zylinder 11 und 21, die zusammen insgesamt 301 Mal nicht gefunden wurden, nicht möglich. Die Fehldetektionen treten in diesen Fall auf, da beide Zylinder einen Radius von 5mm besitzen und auf derselben Achse liegen, weswegen die Kreisdetektion diese nicht in einem Durchlauf finden kann. Werden zwei beliebige Zylinder z_i , z_k mit derselben Achse w nur durch eine Translation entlang w verschoben, so werden ihre Kreise bei der Projektion auf die Ebene übereinander projiziert. In einem solchen Fall wird somit nur ein Kreismittelpunkt c_m für beide Zylinder berechnet. Somit werden nur für einen der beiden Zylinder etwa z_i , die Zylinderendpunkte berechnet. Im Anschluss werden seine Punkte aus den Eingabepunkteset der Kreisdetektion entfernt und es wird erneut eine Kreisdetektion auf den verbliebenen Punkten berechnet. Im Optimalfall findet diese zweite Kreisdetektion ebenfalls c_m für den verbleibenden Zylinder z_k . Wurde jedoch z_i etwa aufgrund eines Fehlers in seiner Achse nicht perfekt detektiert, so können in dem Kreiseingabeset Punkte von z_i verbleiben. Dies kann dazu führen, dass erneut z_i anstatt z_k detektiert wird. Da z_i bereits einmal gefunden wurde, wird die erneute Detektion gelöscht und aufgrund fehlender Neudetektionen keine weitere Kreissuche begonnen. Dieser Fehler sollte in späteren Implementationen behoben werden. Insgesamt treten somit nur 8 Fehldetektionen auf, die nicht durch dieses Problem hervorgerufen wurden.

In den folgenden Abschnitten werden die einzelnen Abweichungen der gefundenen Zylinder noch einmal genauer analysiert und auf einige der bereits erwähnten Probleme genauer eingegangen. Auf den simulierten Daten ist ein grober Fehler für einen Randfall aufgetreten. Um noch robustere Ergebnisse zu bekommen, sollte dieser in zukünftigen Implementierungen behoben werden.

5.3.2 Auswertung Zylinderachsendetektion auf rauschfreier Punktwolke

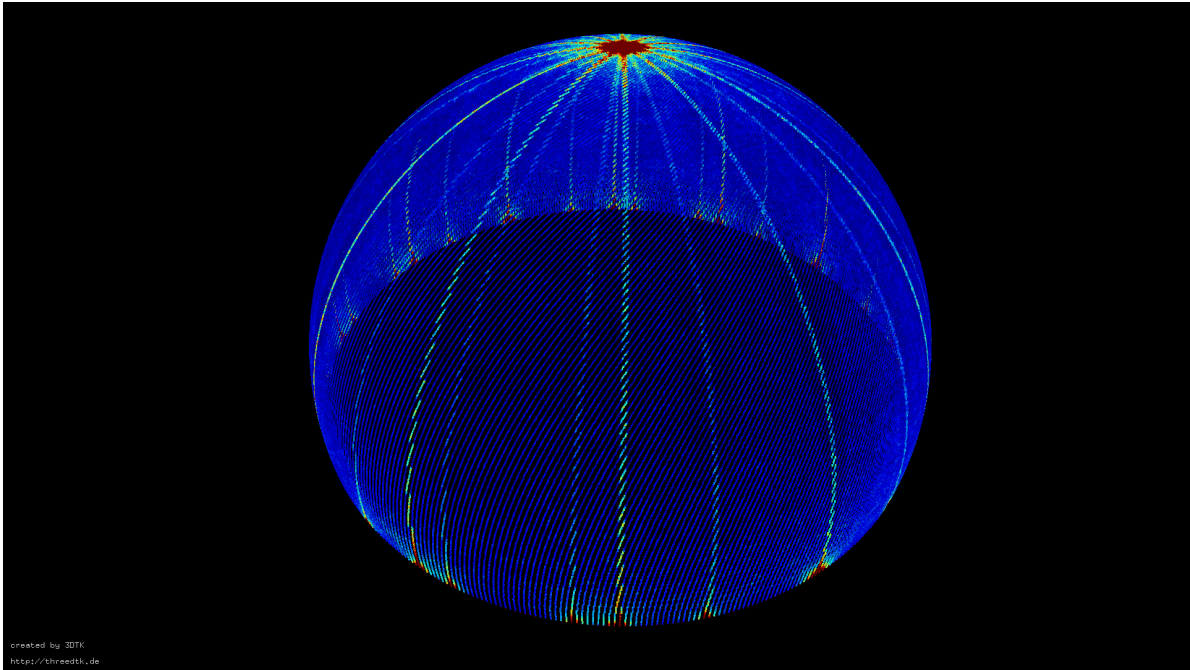


Abbildung 5.20: Darstellung des Akkumulators zur Detektion der Zylinderachsen, wobei blau nicht inkrementiert und rot maximal akkumulierte Zellen beschreibt

Nachdem eine grobe Auswertung in detektiert und falsch detektiert stattgefunden hat, soll nun in diesem Abschnitt die Zylinderachsendetektion genauer betrachtet werden. Dabei entspricht die Anzahl der detektierten Achsen nicht gleich der Anzahl der Zylinder, da sich etwa alle stehenden Zylinder eine Achse mit den Wert $[0 \ 0 \ 1]$ teilen. Auch die Anzahl der Achsen im Vergleich zu den Maxima im Akkumulator sind verschieden. Da die halben Zylinder alle senkrecht zu den stehenden Vollzylindern liegen, besitzen sie im Hough-Raum jeweils zwei Maxima auf der gegenüberliegenden Seite der Halbkugel. So besitzen etwa die fünf Halbzylinder mit dem Radius 7,5mm 10 mögliche Achsen die detektiert werden können. In Abbildung 5.20 wird der Hough-Raum durch den Akkumulator mit seinen Maximum dargestellt. Man kann in der Abbildung gut erkennen, dass das Maxima der stehenden Zylinder, siehe rote Punkte auf der Oberseite der Halbkugel, deutlich größer ist als die Maxima der liegenden am Rande der Sphäre. Dies liegt daran, dass insgesamt 6 Zylinder für die Achse $[0 \ 0 \ 1]$ voten, während meist ein Halbzylinder für zwei Maxima votet.

Nicht nur im Akkumulator kann man die doppelten Maxima erkennen, sie zeigen sich natürlich auch in den verschiedenen Werten der Zylinderachsen und müssen für die Auswertung etwa für die Berechnung eines Mittelwertes in Tabelle 5.17 passend gespiegelt werden.

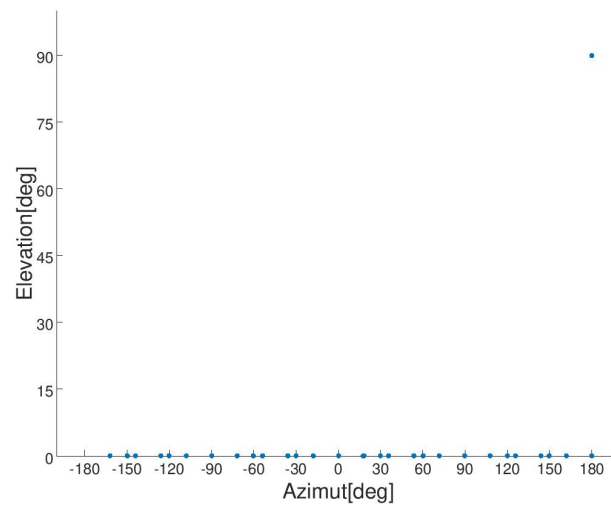
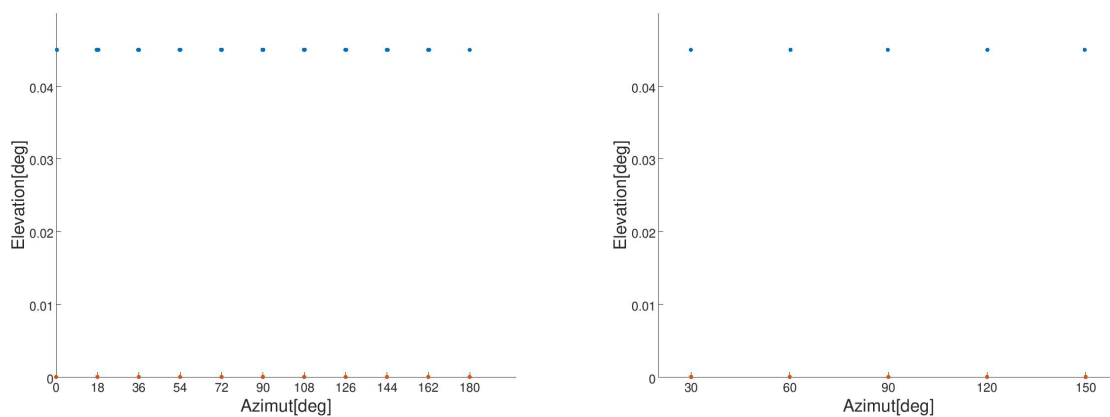


Abbildung 5.21: Darstellung der sphärischen Achsen aller Zylinder



(a) Achsen der liegenden Zylinder mit Radius 5mm (blau:Ist, rot: Soll) (b) Achsen der liegenden Zylinder mit Radius 7.5mm (blau:Ist, rot: Soll)

Abbildung 5.22: Darstellung der Zylinderachsen in sphärischen Koordinaten

In 5.21 werden die normalisierten Achsen in sphärischen Koordinaten dargestellt, um nicht nur die Verteilung der Achsen, sondern gleichzeitig auch die Winkel zwischen diesen direkt betrachten zu können. Während die Achse der stehenden Zylinder $[-7,8540e-04 \ 9,6183e-20 \ 1,0000e+00]$ bei einem Azimut 180° und Elevation 90° liegt, besitzen die Achsen der liegenden Halbzylinder eine Elevation von etwa 0° . Im diesem Wertebereich sind jedoch statt der 14 einzigartigen Achsen 28 aufgetragen, wobei Zylinder 11 und 21, sowie 8 und 16 dieselbe Achse besitzen. Grund sind auch hier die doppelten Maxima des Hough-Raumes.

Betrachtet man die Achsen der liegenden Zylinder mit Radius 7,5 mm oder analog dazu alle halben liegenden Zylinder mit Radius 5 mm einzeln, so kann man auch die Winkeldifferenz von 30° beziehungsweise 18° zwischen den einzelnen Achsen gut erkennen, da die Winkeldifferenz in der Ebene gleich der Winkeldifferenz im Azimut entspricht (siehe 5.22). Für eine übersichtlichere Darstellung wurden alle Achsen, mit negativen Azimut in Abbildung 5.22 gespiegelt und auf die passende gegenüberliegende Achse im Akkumulator gelegt. Es wurde keine komplette Invertierung genutzt, da der Hough-Raum als halbe Sphäre dargestellt wird und somit die dritte Koordinate der Zylinderachse stets positiv ist, da sie auf der ersten Reihe über den Äquator des Akkumulators liegt. Da in diesem Fall $\Delta\phi = 0,00079^\circ$ ist der Fehler sehr klein und vernachlässigbar.

Die graphische Darstellung der Achsen aus Abbildung 5.22 zeigt bereits gut, dass die detektierten Zylinderachsen im richtigen Wertebereich liegen. Für eine genauere Analyse müssen jedoch die Werte der einzelnen Zylinderachsen betrachtet werden. Diese sind in Tabelle 5.19 einzeln aufgetragen. Es ist möglich alle detektierten Achsen direkt wiederzugeben, anstatt den Mittelwert zu betrachten, da für jeden Zylinder minimal eine beziehungsweise maximal drei Achsen detektiert werden. Da für die stehenden Zylinder 1-5 dieselbe Achse detektiert wurde, werden diese in der Tabelle zusammengefasst. Die liegenden Zylinder hingegen besitzen alle mindestens zwei detektierte Achsen, die sich im Akkumulator gegenüber liegen. Neben den detektierten Achsen wurden auch die Fehler derselben aufgetragen. Um die Genauigkeit der Achsen besser abschätzen zu können wurde, neben der fehlerfreien, perfekten Achse auch der Winkel zwischen dieser und der Detektierten berechnet, wobei der maximale Fehler des Winkels etwa $0,51^\circ$ beträgt. Die Differenzen zwischen detektierter und echter Achse sind somit klein.

Auch die beispielhaft dargestellten, relativen Winkel in 5.18, zeigen gut, dass nur ein kleiner Fehler auf den Achsen liegt. Betrachtet man die Winkeldifferenz zwischen den Zylindernachbarn im Kreis, so ergibt sich zwar nie eine exakte Differenz von 30° , jedoch ist der Fehler sehr klein und spiegelt gut die absoluten Fehler der Zylinderachsen wieder.

Zylinder-nachbarn	Mittlere Zylinderachse 1	Mittlere Zylinderachse 2	Winkel [$^\circ$]
Zyl. 6 - 7	[0,865826 -0,500338 0,000785]	[0,499590 -0,866256 0,000785]	29,954297
Zyl. 7 - 8	[0,499590 -0,866256 0,000785]	[0,000460 -0,999995 0,000785]	30,005269
Zyl. 8 - 9	[0,000460 -0,999995 0,000785]	[-0,500316 -0,865838 0,000785]	29,947625
Zyl. 9 - 10	[-0,500316 -0,865838 0,000785]	[-0,865800 -0,500382 0,000785]	30,048238

Tabelle 5.18: Winkeldifferenz zwischen den Zylinderachsennachbarn der Zylinder 6-10 (Sollwert ist 30°)

Insgesamt lässt sich die Achsendetektion auf den simulierten Daten trotz der halben Zylinder als erfolgreich bewerten. Die detektierten Achsen besitzen einen nur sehr kleinen Fehler und die Zylinderachsendetektion ist, wie bereits erwähnt, nicht für die vielen Fehldetektionen von Zylinder 11 und 21 verantwortlich. Der hier entscheidend limitierende Faktor, sofern keine großen Ebenen in der Szene liegen, ist hingegen die Kreisdetektion. Eine Auswertung dieser findet im nächsten Abschnitt statt.

Zylinder	Detektierte Achsen	Perfekte Achse	Fehlerwinkel[°]
Zyl. 1-5	[-0,000785 0,000000 1,000000]	[0,000000 0,000000 1,000000]	0,000000
Zyl. 6	[-0,863390 0,504537 0,000785]; [0,866549 -0,499091 0,000785]	[0,866019 -0,500011 0]	0,300572; 0,083651
Zyl. 7	[-0,501816 0,864974 0,000785]; [0,496361 -0,868116 0,000785]	[0,500011 -0,866019 0]	0,128522; 0,243678
Zyl. 8	[0,003148 -0,999995 0,000785]; [0,003148 0,999995 0,000785]	[0,000000 -1,000000 0]	0,181185; 0,181185
Zyl. 9	[-0,501816 -0,864974 0,000785]; [0,496361 0,868116 0,000785]	[-0,500011 -0,866019 0]	0,128522; 0,243678
Zyl. 10	[-0,863390 -0,504537 0,000785]; [0,866549 0,499091 0,000785]	[-0,866019 -0,500011 0]	0,300572; 0,083651
Zyl. 11	[-1,000000 0,000000 0,000785]; [0,999980 0,006296 0,000785]	[1,000000 0,000000 0]	0,000000; 0,362371
Zyl. 12	[-0,950862 -0,309616 0,000785]; [0,948893 0,315596 0,000785]; [0,952792 0,303623 0,000785]	[0,951056 0,309018 0]	0,026597; 0,401829; 0,327141
Zyl. 13	[-0,808276 -0,588803 0,000785]; [0,804553 0,593880 0,000785]; [0,811967 0,583703 0,000785]	[0,809028 0,587770 0]	0,091797; 0,437740; 0,289782
Zyl. 14	[-0,586256 -0,810125 0,000785]; [0,591345 0,806418 0,000785]	[0,587770 0,809028 0]	0,126616; 0,260621
Zyl. 15	[-0,306621 -0,951831 0,000785]; [0,300622 0,953743 0,000785]; [0,312607 0,949882 0,000785]	[0,309018 0,951056 0]	0,158545; 0,507382; 0,224168
Zyl. 16	[0,003148 -0,999995 0,000785]; [0,003148 0,999995 0,000785]	[0,000000 1,000000 0]	0,181185; 0,181185
Zyl. 17	[-0,306621 0,951831 0,000785]; [0,312607 -0,949882 0,000785]	[-0,309018 0,951056 0]	0,158545; 0,224168
Zyl. 18	[-0,586256 0,810125 0,000785]; [0,591345 -0,806418 0,000785]	[-0,587770 0,809028 0]	0,126616; 0,260621
Zyl. 19	[-0,808276 0,588803 0,000785]; [0,811967 -0,583703 0,000785]	[-0,809028 0,587770 0]	0,091797; 0,289782
Zyl. 20	[-0,950862 0,309616 0,000785]; [0,952792 -0,303623 0,000785]	[-0,951056 0,309018 0]	0,026597; 0,327141
Zyl. 21	[-1,000000 0,000000 0,000785]; [0,999980 0,006296 0,000785]	[-1,000000 0,000000 0]	0,000000; 0,362371

Tabelle 5.19: Detektierte Zylinderachsen und Fehlerwinkel

5.3.3 Auswertung Kreisdetektion auf rauschfreier Punktwolke

Der entscheidend limitierende Faktor bei der Detektion von Teilzylindern, hier speziell Halbzyklindern, ist gegeben durch die Kreisdetektion. Ihre Validierungsparameter müssen stark genug gewählt werden, um Kreisdetektionen in zufälligen Punkten zu vermeiden, aber gleichzeitig flexibel genug sein, um auch Halbkreise zu finden, auf denen deutlich weniger Punkte liegen als auf einem Vollkreis.

In dem gesampelten Beispielmodell besitzen die halben, liegenden Zylinder mit der Länge 30mm eine deutlich größere Mantelfläche, als etwa der ganze, stehende Zylinder mit Höhe 5mm und Radius 5mm. Aus der Anzahl der detektierten Zylinder in Tabelle 5.17 wird ersichtlich, dass dieser kleine stehende Zylinder, der einzig nicht detektierte ist. Die Schwierigkeit des stehenden Zylinders ist, dass er aufgrund seiner geringen Größe kaum Punkte auf der Mantelfläche besitzt. Gerade einmal 840 Punkte liegen auf ihm, wobei die Zylinderkreisfläche für die Detektion nicht genutzt wird, sondern nur die Punkte auf der Mantelfläche. Damit sind für die Detektion des Zylinders gerade einmal etwa 560 Punkte vorhanden. Diese werden zusätzlich aufgrund von Fehlern in der sphärischen Normalenberechnung nicht vollständig für die Kreisdetektion genutzt. Gerade Normalen nahe des Kreisflächenrandes können einen großen Fehler aufweisen. Der entscheidende Faktor der schließlich dazu führt, dass der Zylinder nicht detektiert wird, ist die Validierungsfunktion der Kreisdetektion. Für halbe Zylinder müssen mindestens 25% aller Kreissegmente einen Punkt beinhalten. Die Anzahl der Kreissegmente ist dabei gegeben durch:

$$n_{seg} = \text{ceil}\left(\frac{2 \cdot \pi \cdot r'}{b}\right) = \text{ceil}\left(\frac{2 \cdot \pi \cdot 5}{0,016}\right) = 1964 \quad . \quad (5.1)$$

Es müssen somit mindestens 25% aller 1964 Segmente, also 491, einen Punkt beinhalten. Dies sind gerade einmal 69 Punkte weniger, als die eigentlichen 560 Punkte, die im optimalen Fall alle auf die Ebene projiziert werden, wobei alle Punkte in unterschiedlichen Kreissegmenten liegen müssen. Kommt es jedoch zu kleineren Fehlern in der sphärischen Normalenberechnung, so werden nicht alle Mantelpunkte auf die Ebene projiziert, wodurch der Kreis des Zylinders nicht positiv validiert werden kann. Eine Anpassung der Schrittweite der einzelnen Kreissegmente kann helfen dieses Problem zu umgehen, jedoch kann dies zu einer Erhöhung der Falschdetektionen führen. In dem Spezialfall des in Abbildung 6.3 beschriebenen Modells, kann es dazu kommen, dass anstatt der 6 im Kreis angeordneten, stehenden Zylinder ein Zylinder mit Radius 40 detektiert wird. Deshalb wird auf eine Reduzierung der Kreissegmente in der Validierungsfunktion verzichtet.

Betrachtet man den Mittelwert der detektierten Radii der Vollzylinder, im Vergleich zu dem der Halbzyklinder, dargestellt in der Tabelle 5.20 so kann zumindest ein kleiner Unterschied in den Fehlern sowie der Standardabweichung erkannt werden. Der Mittlere Radius der liegenden Zylinder mit Radius 5mm ist mit 4,9905mm minimal kleiner, als der detektierte Radius von 4,9992mm der ganzen Zylindern und besitzt somit einen um 0.00870mm größeren Fehler. Dies ist jedoch im Vergleich zur Standardabweichung beinahe vernachlässigbar, denn diese ist bei den liegenden Zylindern mit Radius 5mm 0,019666mm und somit deutlich größer, als die der Stehenden mit 8,0149e-05. Auch die Halbzyklinder mit Radius 7,5mm weisen eine deutlich erhöhte

Standardabweichung auf. Trotzdem ist die Abweichung zwischen echten und detektierten Radii im Falle der Simulationsdaten sehr gering.

Zylinder	Radius [mm]	Mittlerer Radius [mm]	Standardabweichung [mm]
Ganze Zylinder Mitte	5	4,9992	8,0149e-05
Halbe Zylinder	7,5	7,4994	0,014368
Halbe Zylinder	5	4,9905	0,019666

Tabelle 5.20: Mittlere Radii und Standardabweichung Vergleich

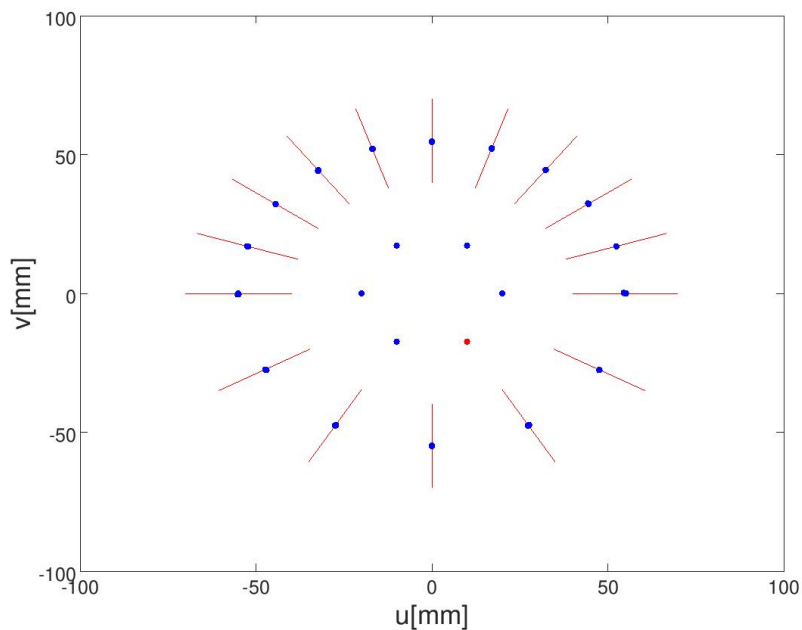


Abbildung 5.23: Darstellung detektierter Zylinderachsenmittelpunkte (blau) und projizierte Zylinderachsen (rot)

Neben den Radii sind für die Kreisdetektion vor allem auch die Positionen der Zylinder interessant. Hierfür wird der detektierte Kreismittelpunkte zurück in das dreidimensionale Koordinatensystem transformiert, um einen Punkt auf der Zylinderachse zu erhalten. In Abbildung 5.23 werden die Punkte auf allen Zylinderachsen auf die Modellebene projiziert, dargestellt. Dabei stellen die roten Linien, beziehungsweise Punkte, die perfekten Zylinderachsen, der einzelnen Zylinder dar. Die blauen Punkte entsprechen den detektierten Punkten auf den gefundenen Zylinderachse. Da die Achsen der stehenden Zylinder projiziert auf die Ebene einen Punkt ergeben, werden sie von den blauen detektierten Zylinderachsenmittelpunkte verdeckt, mit Ausnahme eines Punktes, der den kleinsten, nicht detektierten, stehenden Zylinder darstellt. Für eine gute Detektion müssen diese möglichst genau auf den eigentlichen Achsen

liegen. Visuell betrachtet liegen alle blauen Punkte auf den ihrer jeweiligen Achse, durch die kleine Streuweite wirken sie zudem meist wie ein einzelner Punkt. In Tabelle 5.21, werden noch einmal die mittleren Punkte der Achsen, ihre mittlere Fehler, sowie die darauf beruhende Länge des Zylinders berechnet.

Zylinder	Mittlerer Punkt auf Achse [mm]	Mittlere Fehler- distanz[mm]	Distanz Standard- abweichung [mm]	Zylinderlänge/ Fehler[mm]
Zyl. 1	[-9,999 -17,320 4,276]	0,000500	0,000000	7,687/-2,313
Zyl. 2	[-20,000 0,000 7,637]	0,000000	0,000000	14,267/-0,733
Zyl. 3	[-10,001 17,321 9,738]	0,001118	0,000000	17,868/-2,132
Zyl. 4	[9,999 17,321 12,668]	0,001118	0,000000	22,978/-2,022
Zyl. 5	[20,000 0,000 14,914]	0,000001	0,000008	29,402/-0,598
Zyl. 6	[47,618 -27,510 0,002]	0,016817	0,008367	28,704/-1,296
Zyl. 7	[27,413 -47,503 0,004]	0,031110	0,011804	28,794/-1,206
Zyl. 8	[0,011 -54,877 0,001]	0,053032	0,004220	28,622/-1,378
Zyl. 9	[-27,454 -47,566 0,003]	0,061595	0,008217	28,667/-1,333
Zyl. 10	[-47,348 -27,334 0,001]	0,016116	0,012676	29,201/-0,799
Zyl. 11	[55,055 0,006 -0,001]	0,087294	0,007348	28,576/-1,424
Zyl. 12	[52,458 17,045 0,001]	0,008394	0,000769	28,910/-1,090
Zyl. 13	[44,534 32,355 0,001]	0,018713	0,005937	28,719/-1,281
Zyl. 14	[32,369 44,552 0,003]	0,063396	0,002060	28,447/-1,553
Zyl. 15	[16,964 52,205 0,003]	0,037086	0,001924	28,801/-1,199
Zyl. 16	[-0,001 54,764 0,003]	0,045609	0,000607	28,418/-1,582
Zyl. 17	[-16,906 52,137 0,119]	0,035108	0,006759	28,600/-1,400
Zyl. 18	[-32,320 44,438 0,126]	0,063229	0,012007	28,439/-1,561
Zyl. 19	[-44,394 32,255 0,000]	0,019965	0,001930	28,988/-1,012
Zyl. 20	[-52,297 16,992 0,004]	0,078305	0,002407	28,500/-1,500
Zyl. 21	[-55,079 -0,000 0,040]	0,002329	0,021854	28,499/-1,501

Tabelle 5.21: Mittlere Punkt auf der Zylinderachse sowie die mittlere Fehlerdistanz und Standardabweichung

Es ist gut zu erkennen, dass der mittlere Fehler der Distanz, mit einem Höchstwert von 0,087294mm, sehr gering ist. Auch der Unterschied zwischen den einzelnen Fehlerdistanzen, beschrieben durch die Standardabweichung besitzt gerade einmal einen Maximalwert von 0,021854mm. Die Länge der Zylinder ist aufgrund des gewählten Ansatzes zur Berechnung der Zylinderenden immer geringer als der eigentliche Wert. Für diese Berechnung wird das Eingabepunkteset der Kreisdetektion verwendet, welches nur noch die Punkte beinhaltet, deren Normalen für die Zylinderachse gevoftet haben. Da die Mantelpunkte nahe der Zylinderkante aufgrund ihrer fehlerhaften Normalen meist nicht für die Zylinderachse gevoftet haben, werden diese für die Zylinderenddetektion nicht beachtet.

Zum Schluss soll noch einmal festgehalten werden, dass die Kreisdetektion funktioniert, sofern keine Randbedingungen wie zwei Zylinder mit gleicher Achse und Radius, nur getrennt durch einen Verschiebung entlang ihrer Achse, auftreten. Die Wahl der Validierungsparameter, ist wichtig, da sie Ausreißer minimiert, gleichzeitig setzt sie jedoch auch harte Grenzen bei Zylindern mit einer geringen Anzahl an Mantelflächenpunkten.

5.3.4 Bewertung Zylinderdetektion für halbe Zylinder

Auch wenn mit dieser Arbeit eine allgemeingültige Lösung für die Detektion von Zylindern gesucht wird, werden an diesem Beispiel doch Grenzen deutlich. Für die Detektion von Teilzylindern kann festgehalten werden, dass der Algorithmus zwei große Einschränkungen aufweist. Die erste ist gegeben durch die Bedingungen der Validierungsparameter der Kreisdetektion. Die Standard-Parameter müssen teilweise für die verschiedenen Punktwolken angepasst werden, insbesondere wenn Zylinder mit nur wenig Mantelflächenpunkten oder viele Ausreißer vorhanden sind. In manchen Fällen ist es nur sehr schwer möglich ein optimales Set zu finden, vergleiche hierzu vorausgegangenes Problem der stehenden Zylinder. Des Weiteren sollte der Algorithmus erweitert werden, sodass auf einer Achse mehrere Zylinder mit gleichem Radius nur getrennt durch eine Translation auf dieser ebenfalls fehlerfrei detektiert werden. Es sei jedoch auch festgehalten, dass die Detektion der restlichen Zylinder gut ist und sowohl die Position, der Radius als auch die Achse nur einen sehr kleinen Fehler aufweisen. Die Detektion von Halbzylindern ist auf simulierten Daten kein Problem.

5.3.5 Generierung realer Scan und Kategorisierung der detektierten Zylinder

Um anschließend die Detektion von Teilzylindern auch auf realen Daten zu testen, wurde das Modell (dargestellt in Abbildung 6.3) 3D-gedruckt. Anschließend wurde das Modell mithilfe eines scanControl 2900-25 von Micro-Epsilon [4] gescannt. Um aus den 2D-Daten eine 3D-Punktwolke zu erstellen wurde ein Kuka KR 16 verwendet, um über die Trajektorie des Industrieroboters zeilenweise das Modell aus zwei verschiedenen Richtungen gescannt, sodass zwei sich überlagernde Punktwolken entstanden. Um diese zu Matchen wurde das Toolkit 3DTK verwendet [1]. Dieses besitzt neben einer Ebenendetektion auch die Möglichkeit zwei Punktwolken mithilfe von ICP zu Matchen und als eine linkshändige Punktwolke, dargestellt in Abbildung 5.24, zu exportieren.

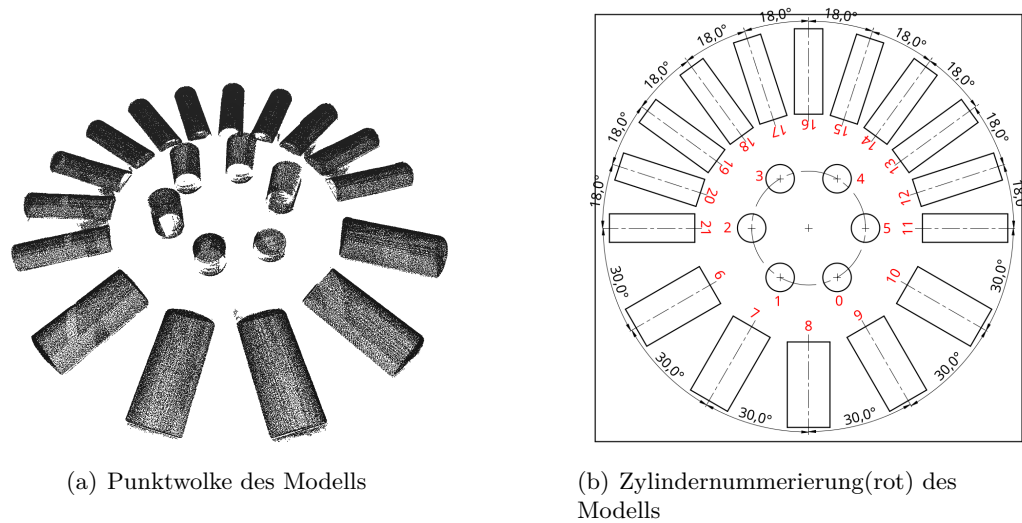
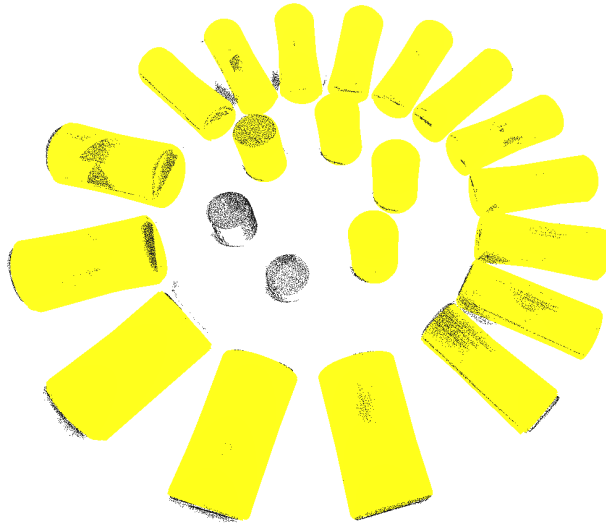


Abbildung 5.24: Punktwolke und Zylinder Nummerierung des Modells 3

Im Gegensatz zu dem Scan von Abschnitt 5.2 fand keine weitere Nachbearbeitung statt. Somit befinden sich in diesem große Lücken, etwa durch Schatten, die man gut in Abbildung 5.24 auf den stehenden Zylindern erkennen kann. Gleichzeitig sind jedoch die Kanten der eigentlichen Zylinder erhalten geblieben, so dass die Zylinderkreisfläche nicht Kugelförmig sondern als Ebene dargestellt wird.

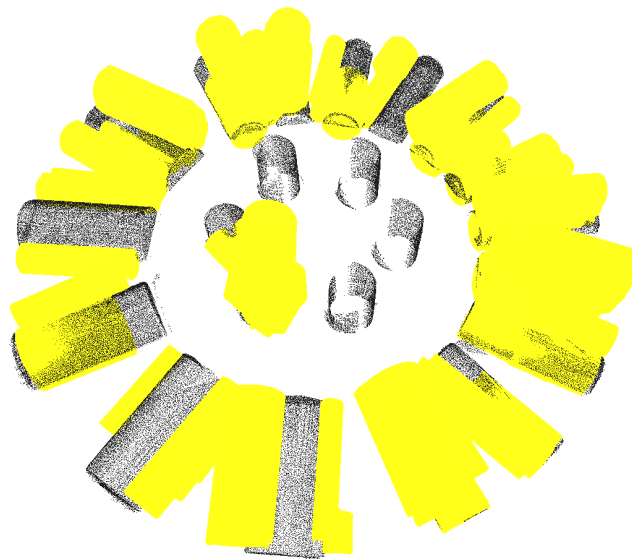
Da aus Abschnitt 5.2 bekannt ist, welche Auswirkungen Ebenen auf die Zylinderachsen-detektion haben, wurde die Bodenplatte nahezu vollständig aus dem Scan entfernt. Insgesamt besteht die Punktwolke aus 238788 Punkten. Erneut werden für die Auswertung anstatt einer Detektion je 1000 Detektionen betrachtet, wobei die mittlere Dauer einer Detektion bei 37,5s liegt (verwendete CPU: AMD Ryzen 5 2600X). Für die folgenden Absätze wird die Zylinder Nummerierung dargestellt in der linken Abbildung von 5.24 verwendet.

In 1000 Durchläufen wurden statt der erwarteten 22000, beziehungsweise 21000 da der kleinste Vollzylinder bereits auf simulierten Daten nicht detektiert wurde, insgesamt 20796 Zylinder gefunden. Im Gegensatz zu Abschnitt 5.2 ist eine direkte Aufteilung der Zylinder in korrekt und falsch, beziehungsweise fehlend detektierte Zylinder nur sehr schwer möglich, da viele Falschdetektionen ähnliche Achsen wie die Halbzylinder haben. Da aufgrund der Real-daten auch keine absoluten Werte existieren, werden die Zylinder deren Zylinderachsendifferenz zu ihren Nachbarn am Besten den relativen Sollwinkel entsprechen herausgefiltert. Mithilfe ihrer Mediane werden die mittleren Anfangs, sowie Endpunkte der Zylinderachsen bestimmt und als Referenz zum Aufteilen der Zylinder verwendet. Anschließend werden alle Zylinder deren Anfangs- beziehungsweise Endpunkte eine kleinere Distanz als t_{dis} zu den jeweiligen Referenzpunkten aufweisen herausgefiltert. Für eine maximale Distanz von 5.0 mm werden insgesamt 19376 Zylinder aus den 20796 Zylindern herausgefiltert. Dies sind etwas mehr als 93% aller gefundenen Zylinder. Sie werden im Folgenden als detektiert bezeichnet. Es wurden somit bezogen auf die 21000 Zylinder insgesamt 92,3% aller Zylinder gefunden.



created by 3DTK
<http://threadtk.de>

Abbildung 5.25: Erfolgreiche Detektionen (gelb) in realer Punktvolke



created by 3DTK
<http://threadtk.de>

Abbildung 5.26: Fehlerhafte Detektionen in realer Punktvolke

Die restlichen 6,8% sind jedoch nicht reine Falschdetektionen. Insbesondere bei den halben liegenden Zylindern mit Radius 5mm handelt es sich oft um unvollständig detektierte Zylinder, deren Zylinderhöhe zu kurz bestimmt wurde oder die eine leichte Translation erfahren haben. Etwa 5% der 21000 Zylinder, also 1050 Detektionen sind komplette Falschdetektionen, da sie zwischen zwei Zylindern liegen oder auf die restlichen Punkte der stehenden Zylinder gematcht wurden. Alle nicht einsortierten Zylinder sind in Abbildung 5.26 dargestellt. Für eine Reduktion der Falschdetektionen müssen die Parameter der Kreisvalidierungsfunktion angepasst werden. Etwa durch eine feinere Kreisschrittauflösung oder durch Erhöhung der Anzahl an Segmenten, auf denen ein Punkt liegen muss. Dies ist im gegebenen Fall der Realdaten jedoch nur sehr schwer möglich, da es sich bei den meisten Zylindern um halbe Zylinder handelt. Eine Einengung der Parameter würde somit die Anzahl der Falschdetektionen verringern jedoch gleichzeitig die Fehldetektionsrate erhöhen.

Zylinder [Radius; Höhe]	Anzahl	Mittlerer Radius [mm]	Mittlere Achse
Zyl. 0-1 [5mm; [5, 10]mm]	-	-	-
Zyl. 2-5 [5mm; [15,20, 25, 30]mm]	3997	4,977088	[0,001380 0,999189 0,006357]
Zyl. 6 [7,5mm; 30mm]	993	7,427172	[-0,493197 -0,001885 0,869653]
Zyl. 7 [7,5mm; 30mm]	999	7,526029	[-0,850309 0,007853 0,525317]
Zyl. 8 [7,5mm; 30mm]	996	7,426477	[0,999116 -0,008590 0,014987]
Zyl. 9 [7,5mm; 30mm]	1000	7,686630	[0,878575 -0,004868 0,476879]
Zyl. 10 [7,5mm; 30mm]	1000	7,757408	[0,512793 -0,002961 0,858181]
Zyl. 11 [5mm; 30mm]	927	5,234428	[0,002132 -0,004140 0,999379]
Zyl. 12 [5mm; 30mm]	989	5,108334	[-0,302530 0,000483 0,952829]
Zyl. 13 [5mm; 30mm]	994	5,076445	[-0,571366 0,004360 0,820178]
Zyl. 14 [5mm; 30mm]	831	4,905990	[-0,799127 0,010327 0,599457]
Zyl. 15 [5mm; 30mm]	877	4,867855	[-0,943658 0,008873 0,327434]
Zyl. 16 [5mm; 30mm]	978	4,897739	[0,998687 0,025507 0,015339]
Zyl. 17 [5mm; 30mm]	926	4,979150	[0,955972 -0,007939 0,289266]
Zyl. 18 [5mm; 30mm]	998	4,946013	[0,823721 -0,008050 0,566299]
Zyl. 19 [5mm; 30mm]	989	4,805956	[0,601837 -0,005098 0,798163]
Zyl. 20 [5mm; 30mm]	993	4,869633	[0,321836 0,000422 0,946461]
Zyl. 21 [5mm; 30mm]	889	4,909966	[0,017559 -0,002393 0,998978]

Tabelle 5.22: Detektierte Zylinder, sowie mittlerer Radius und mittlere Zylinderachse

In Tabelle 5.22 sind die einzelnen Zylinder nach der Anzahl ihrer Detektionen aufgeschlüsselt, wobei die stehenden Zylinder in zwei Gruppen zusammengefasst wurden. Die erste Gruppe besteht aus den zwei kleinsten Zylindern 0 mit Höhe 5mm und 1 mit Höhe 10mm, aufgrund der großen Schatten fehlen viele Punkte auf der Manteloberfläche, aus diesem Grund können sie wie schon Zylinder 0 im Simulationsdatenset von der Kreisvalidierungsmethode erkannt werden. Eine Anpassung der Validierungsparameter etwa durch eine Senkung der Schrittweite der Kreisauflösung ist nicht möglich, da dies die Anzahl der Falschdetektionen erhöht. Die zweite Gruppe besteht aus den Zylindern 2, 3, 4 und 5 mit den Höhen 15, 20, 25 und 30mm. Zylinder 3-5 wurden 1000 Mal detektiert, Zylinder 2 wurde nur in drei Durchläufen nicht detektiert. Die Genauigkeit ihrer Achse, sowie Radius und Position ist jedoch sehr ähnlich, weswegen sie im folgenden oft zusammen betrachtet werden.

Auch die halben liegenden Zylinder mit Radius 7,5mm werden noch bei fast jedem Durchlauf detektiert. So wurden Zylinder 9 und 10 1000 Mal gefunden, während Zylinder 6 mit sieben Fehldetektionen am häufigsten nicht detektiert wurde. Somit weisen die halben Zylinder mit einem Radius von 7,5mm eine relativ kleine Fehldetektionsrate auf. Diese steigt jedoch deutlich betrachtet man die halben Zylinder mit Radius 5,0mm, so wurde etwa Zylinder 13 831 Mal detektiert. Auffällig ist jedoch dass die Zylinder auf einer Achse, etwa Zylinder 11 und 21, Detektionsraten von 889 und 927 haben, was immer noch unter dem Mittelwert von 944.636 (beziehungsweise 956 ohne Zylinder 13) Funden liegt, jedoch weitaus weniger kritisch auffällt als bei den Simulationsdaten. Dieses eingangs beschriebene Problem zeigt sich abgeschwächt, da die Maxima im Hough-Raum für die Zylinderachsendetektion deutlich breiter sind, wodurch mehr Achsen getestet werden und eine größere Wahrscheinlichkeit besteht beide Zylinder zu finden.

Da Zylinder 1 und 0 zu wenige Mantelpunkte besitzen, um mit den gewählten Validierungsparametern berechnet zu werden, werden sie in die folgende Auswertung nicht einbezogen. Somit sollten ohne diese zwei Zylinder insgesamt 20000 Zylinder detektiert werden, die gefundene Anzahl von 19376 ist um 624 Zylinder zu gering. Etwas mehr als die Hälfte dieser fehlenden Zylinder wurde als Falschdetektion gewertet, da mindestens einer ihrer Parameter zu große Abweichungen aufzeigt. Allgemein lässt sich aus der Tabelle 5.22 ablesen, dass niedrigere Anzahlen an Mantelflächenpunkten sich negativ auf den Detektionserfolg auswirken. In den folgenden Abschnitten werden die Fehler der einzelnen Parameter analysiert und ihre Verteilung genauer betrachtet.

5.3.6 Auswertung Zylinderachsendetektion auf realen Daten

Da es sich um Realdaten handelt, ist wie auch bei der Aufteilung der Zylinder ein Vergleich der gefundenen Achsen mit absoluten Werten nicht möglich. Stattdessen werden für die Evaluation die Winkel zwischen den detektierten Achsen der Zylinder berechnet, da bekannt ist, dass der Winkel zwischen zwei Halbzylindern mit Radius 5 mm 18° und sonst 30° beträgt.

In der Tabelle 5.23 sind die einzelnen Differenzen der mittleren Winkel sowie deren Fehler zu dem eigentlichen Wert (30° oder 18°) aufgetragen. Dabei beträgt der größte Fehler etwa 2.7° und der minimale -0.053° . Da die Winkeldifferenzen eigentlich für die Achsen in der Ebene bekannt sind, ist es ebenfalls sinnvoll die Winkel der Achsen zu dieser Ebene zu betrachten. Projiziert man

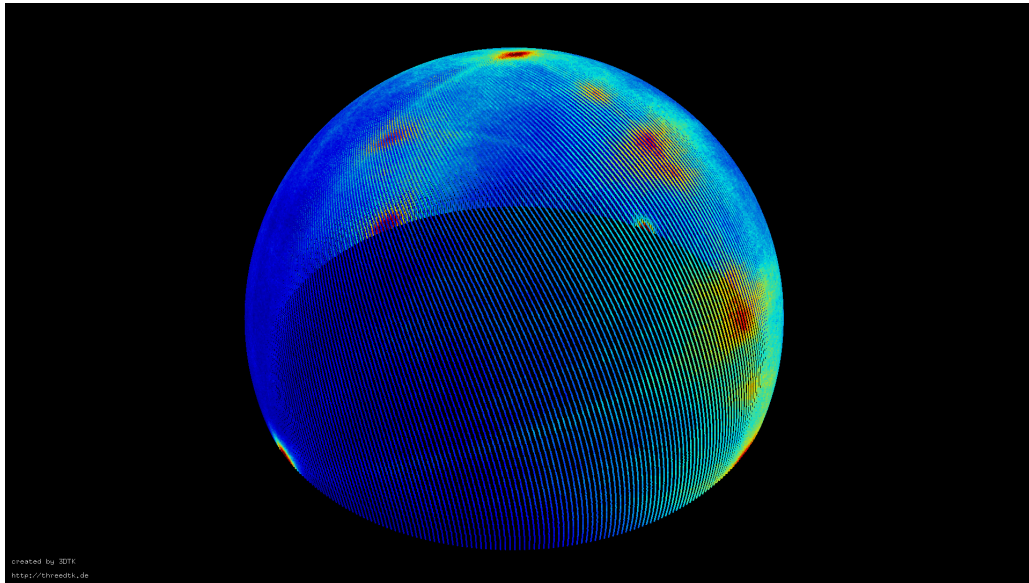


Abbildung 5.27: Darstellung des Akkumulators zur Detektion der Zylinderachsen, wobei blau nicht inkrementiert und rot maximal akkumulierte Zellen beschreibt

die Achsen entlang der mittleren Zylinderachse von 2-5, die eine fehlerbehaftete Ebenennormale wiedergibt und berechnet die Winkeldifferenz so liegt der Fehlerbereich in $[-2,25^\circ, 2,68^\circ]$ mit der kleinsten Abweichung von -0.04° . Der mittlere absolute Fehler beträgt $0,951^\circ$, was nur minimal kleiner ist als der absolute Fehler der nicht projizierten Achsen der $0,952^\circ$ beträgt. Die relativen Winkel sind kein absoluter Indikator für die Fehler der einzelnen Achsen, indizieren jedoch, dass diese zumindest im richtigen Bereich liegen.

Da der absolute Wert der Achse nicht bekannt ist, ist es wichtig die Verteilung der einzelnen Achsen zu betrachten. Dabei kann für eine sehr grobe Analyse bereits die Abbildung des Hough-Raumes (siehe Abbildung 5.27) helfen. Die Punkte auf dem Großkreis repräsentieren die halben liegenden Zylinder, während die stehende Achse am Rande der Sphäre (im Bild links unten) dargestellt wird. Es ist dabei deutlich erkennbar, dass die Maxima unterschiedlich weit streuen und teilweise kleine Maxima, die meist von halben liegenden Zylindern mit Radius 5mm stammen, am Rande der großen Maxima liegen. Diese großen Maxima entstehen etwa durch Halbzylinder mit Radius 7,5mm oder zwei halbe gegenüberliegende Zylinder, die dieselbe Achse besitzen, weswegen deutlich mehr Punkte auf der Mantelfläche im Hough-Raum für diese voten. Liegt nun ein kleines Maximum in einem Randbereich eines großen Maximum, ist es deutlich schwieriger dieses zu detektieren und exakt im Akkumulator zu bestimmen. Dies liegt daran, dass leicht fehlerhafte sphärische Normalen auch für einen Bereich, um das lokale Maximum abstimmen und somit die Detektion einer Zylinderachse mit nur wenig Mantelpunkten beeinflussen können. Die den realen Daten inhärenten Fehler verstärken dieses Problem, da so sphärische Normalen nicht exakt berechnet werden können. Aus diesem Grund ist die Detektion von kleinen Maxima, am Rande von größeren, deutlich schwieriger als etwa das klare Maximum der sechs stehenden Zylinder.

Zylinder	Zylinder Nachbar	Mittlerer Winkel [°]	Winkelfehler [°]
Zyl. 6	Zyl. 7	28,805887	-1,194113
Zyl. 7	Zyl. 8	32,686509	2,686509
Zyl. 8	Zyl. 9	27,755328	-2,244672
Zyl. 9	Zyl. 10	30,707005	0,707005
Zyl. 10	Zyl. 11	30,823775	0,823775
Zyl. 11	Zyl. 12	17,925793	-0,074207
Zyl. 12	Zyl. 13	17,367793	-0,632207
Zyl. 13	Zyl. 14	18,921310	0,921310
Zyl. 14	Zyl. 15	17,850071	-0,149929
Zyl. 15	Zyl. 16	20,625440	2,625440
Zyl. 16	Zyl. 17	16,529179	-1,470821
Zyl. 17	Zyl. 18	17,947110	-0,052890
Zyl. 18	Zyl. 19	18,615572	0,615572
Zyl. 19	Zyl. 20	18,342506	0,342506
Zyl. 20	Zyl. 21	18,042419	0,042419
Zyl. 21	Zyl. 6	30,650542	0,650542

Tabelle 5.23: Mittlerer Winkel zwischen projizierten Zylinderachsen

Betrachtet man die Streuung der einzelnen Achsen so kann man eben dieses Phänomen gut wiedererkennen. In Abbildung 5.28 wird beispielhaft ein Histogramm für die Achse des Zylinders 9 dargestellt. Zylinder 9 ist ein halber liegender Zylinder mit Radius 7,5mm. Er besitzt somit deutlich mehr Punkte als etwa ein liegender Zylinder mit Radius 5mm. Im Histogramm erkennt man gut, dass die Streuweite auf einen sehr kleinen Raum begrenzt ist und eher eine Linienförmige Struktur aufweist. In 5.29 wird hingegen die Achse des Zylinders 17 aufgetragen. Hier ist zwar ein Peak erkennbar jedoch ist dieser deutlich breiter gestreut. Die Achsen des Zylinders 17 besitzen im Vergleich zu Zylinder 9 eine deutlich größere Streuweite. Eine ähnliche Verteilung weisen auch die restlichen detektierten Zylinderachsen auf. Diese sind nicht immer Peak förmig wie Zylinder 17, jedoch sind die Achsen der halben liegenden Zylinder deutlich weiter gestreut.

Betrachtet man das Ergebnis der Zylinderachsendetektion insgesamt, so kann man festhalten, dass die einzelnen Zylinderachsen im wesentlichen gut detektiert wurden. Die Winkel zwischen den einzelnen Zylindern bleiben erhalten. Obwohl es sich um Realdaten handelt, war die Zylinderachsendetektion für halbe Zylinder möglich, die Streuweite der Achsen ist jedoch deutlich größer, gerade bei Halbzylindern mit kleinen Radii.

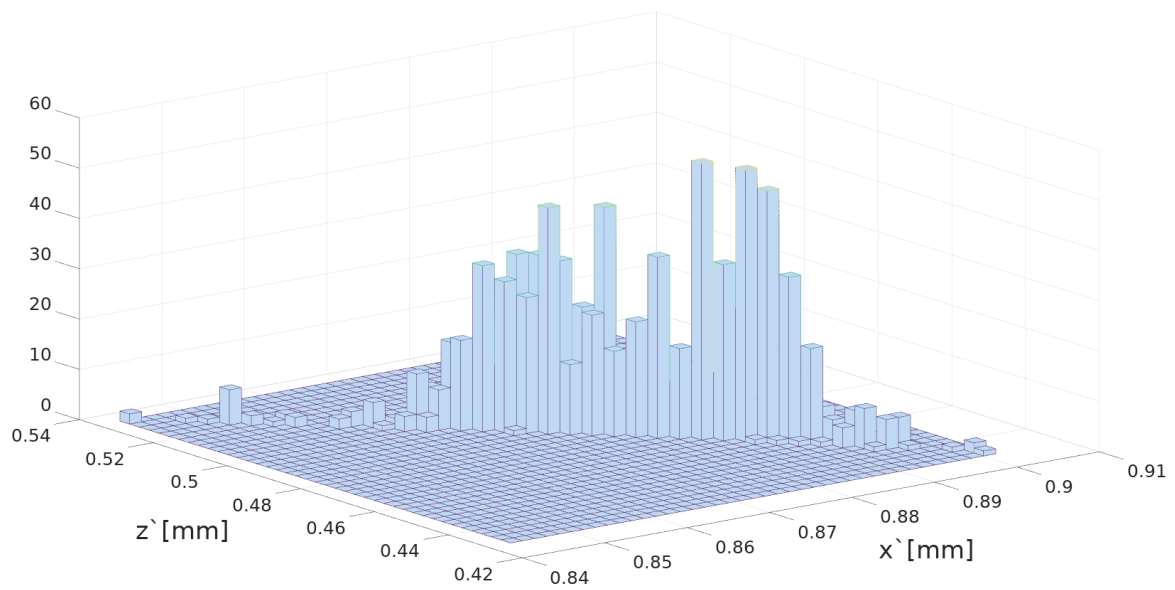


Abbildung 5.28: Histogramm der Abweichung der Soll-Achse des Zylinders 9

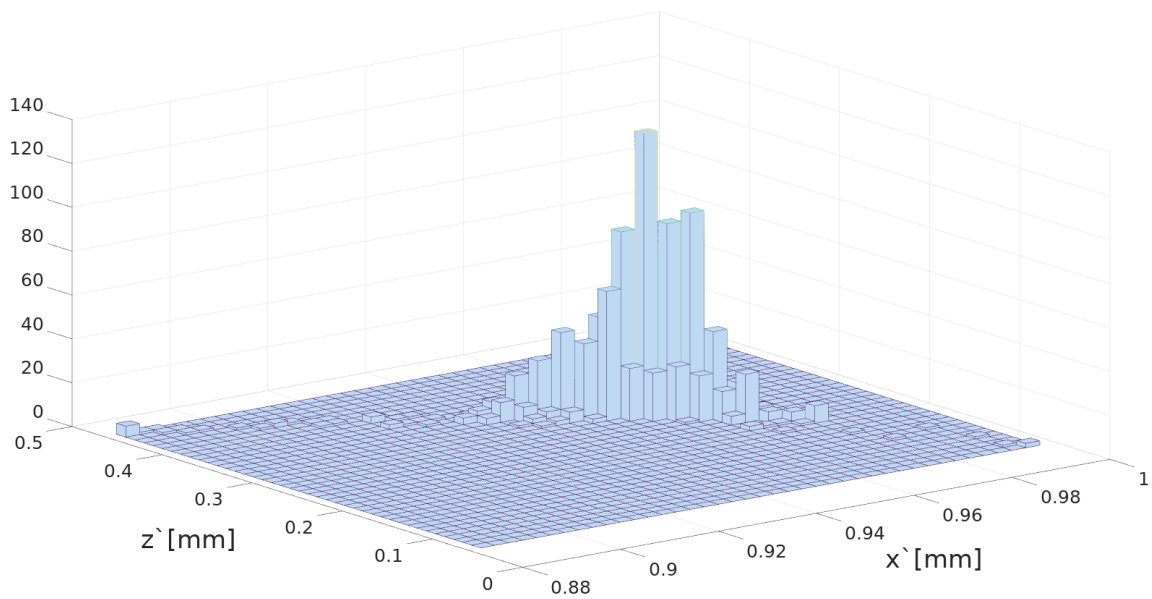


Abbildung 5.29: Histogramm der Abweichung der Soll-Achse des Zylinders 17

5.3.7 Auswertung Kreisdetektion auf realen Daten

Wie bereits in Abschnitt 5.3.5 ist jedoch der entscheidend limitierende Faktor nicht die Zylinderachsendetektion, sondern die Kreisdetektion, sofern keine großen Ebenen vorhanden sind. Die Validierungsparameter, die bestimmen ob ein Punkteset P , welches auf einer Kreisbahn K liegt, ein Kreis darstellt oder nicht, müssen aufgrund der Halbzylinder im Vergleich zu Abschnitt 5.2 angepasst werden. Dies erfolgt in diesem Fall durch eine Senkung des positiven Schwellwerts von 50% auf 25 % aller Kreissegmente, die einen Punkt beinhalten müssen. Aus diesem Grund steigt die Anzahl der Falschdetektionen deutlich, da P auf K häufiger als Kreis erkannt wird, auch wenn es sich bei P nur um ein zufälliges Punkteset handelt. Gleichzeitig besitzen halbe Zylinder, insbesondere bei kleiner Mantelfläche, eine erhöhte Fehldetektionsrate im Vergleich zu Vollzylindern. Zudem haben die detektierten Halbkreise eine deutlich erhöhte Unsicherheit, weswegen die Schwankungen des Kreismittelpunktes sowie des Radius stark zunehmen.

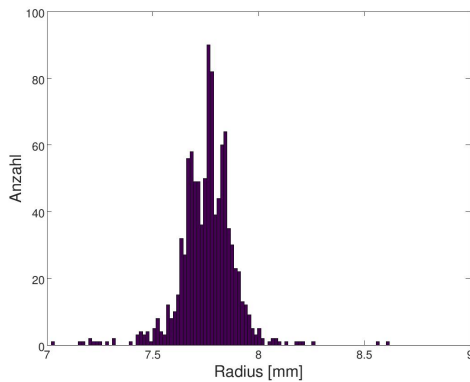
Dieses Phänomen kann gut durch die in Tabelle 5.24 dargestellten Radii, sowie ihrer Standardabweichung beobachtet werden. Während der mittlere Radius der ganzen Zylinder eine Abweichung von 0,022912mm aufweist und eine Standardabweichung von nur 0,071107mm besitzt, weist der mittlere Radius der halben Zylinder mit dem deutlich größeren Radius von 7,5mm einen Fehler von -0.059713mm auf und die Standardabweichung ist etwa 4,6 mal größer, als die der ganzen Zylinder. Betrachtet man die halben Zylinder mit dem Radius 5,0mm so zeigt sich eine noch größere Differenz. Der Fehler des Mittelwertes beträgt hierbei schon 0,094329mm, während die Standardabweichung von 0,432650 bereits um Faktor 6 größer ist wie die der stehenden ganzen Zylinder mit Radius 5mm.

Zylinder	Radius [mm]	Mittlerer Radius [mm]	Standardabweichung [mm]
Ganze Zylinder	5	4,977088	0,071107
Halbe Zylinder	7,5	7,559713	0,329613
Halbe Zylinder	5	4,905671	0,432650

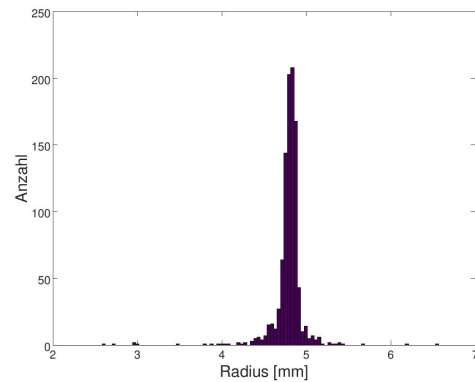
Tabelle 5.24: Mittlere Radii und Standardabweichungen Vergleich

Betrachtet man die Radii mit der geringsten Standardabweichung, sind das im Falle der halben liegenden Zylinder mit Radius 7,5mm Zylinder Nummer 10 und für einen Radius von 5mm Nummer 18. Auffällig ist dabei, dass sowohl Zylinder 10, als auch Zylinder 18 sehr hohe Detektionsquoten von 1000 bzw. 998 aufweisen. In Abbildung 5.30 werden die einzelnen detektierten Radii dieser beiden Zylinder dargestellt. Bereits bei dem Vergleich der Histogramme ist anhand der x-Achse gut zu erkennen, dass Zylinder 18 eine deutlich breitere Verteilung aufweist als Zylinder 10.

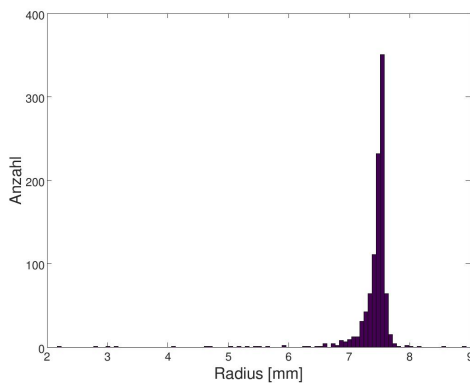
Diese Beobachtung spiegelt sich noch stärker in den Radiusbereichen der einzelnen Zylinder wieder, während Zylinder 10 einen kleinen Bereich von [7,0202 8,6189] besitzt, ist der Wertebereich der Radii von Zylinder 18 mit [1,5610 5,5301] deutlich kritischer. Hingegen ist der Median des kleinen Halbzylinders 18 mit 4,9184mm besser, als der Median des Zylinders 10 mit 7,7651mm. An dem Median kann wiederum gut erkannt werden, dass nur wenig Mantelpunkte



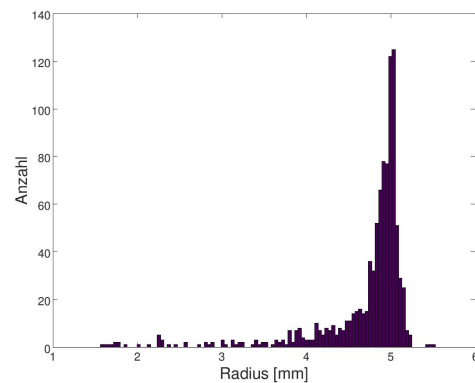
(a) Histogramm der 1000 detektierten Radii des Zylinders 10



(b) Histogramm der 1000 detektierten Radii des Zylinders 18

Abbildung 5.30: Histogramme der Radii von Zylinder 10 (links) und 18 (rechts)

(a) Histogramm der 1000 detektierten Radii des Zylinders 6



(b) Histogramm der 1000 detektierten Radii des Zylinders 15

Abbildung 5.31: Histogramme der Radii von Zylinder 6 (links) und 15 (rechts)

für die Kreisdetektion genutzt werden können und es somit zu einem sehr guten oder keinem Fit kommt. Im Spezialfall der halben Zylinder kann es auch zu einem Fit mit sehr kleinem Radius kommen, der den eigentlichen Zylindermantel tangiert. Letzteres wurde durch die Herabsetzung der Kreisdetektionsvalidierungsparametern möglich. Durch den sehr kleinen Radius von $1,5610$ müssen gerade einmal $0,25 \cdot 395 = 99$ Segmente einen Punkt enthalten, was durch Ausreißer und die Punkte auf der ursprünglichen Mantelfläche des Zylinders vorkommen kann. Neben der visuellen Betrachtung beschreibt die Standardabweichung die Verteilung der detektierten Radii von Zylinder 10 beziehungsweise 18 und somit deren Unterschied sehr gut. Während Zylinder 10 eine Standardabweichung von $0,12821$ besitzt, weist Zylinder 18 mit $0,21373\text{mm}$ eine fast doppelt so große auf.

Die Standardabweichung von Zylinder 18 ist jedoch kleiner als die von Zylinder 6, obwohl dieser ebenfalls einen Radius von 7,5 mm besitzt. Zylinder 6 weist mit 0,42250mm die größte Standardabweichung aller halben Zylinder mit Radius 7,5mm vor. Dies kann gut aus Tabelle 5.25 abgelesen werden, in welcher die einzelnen Mediane, Mittlere Radii und Standardabweichungen der Fehler aufgetragen sind. Die größte Varianz besitzt jedoch Zylinder Nummer 15 mit einer Standardabweichung von 0,585470mm. In Abbildung 5.31 sind aus diesem Grund neben dem Histogramm der gefundenen Radii von Zylinder 6 auch die Radii von Zylinder 15 aufgetragen. In dieser Darstellung kann gut erkannt werden, dass für Zylinder 15 sehr viele kleine und in Einzelfällen signifikant zu kleine Radii gefunden wurden. Trotzdem kann selbst für den am schlechtesten detektierten Zylindern ein deutlicher Peak im eigentlichen Radiusbereich gefunden werden.

Zylinder	Radius [mm]	Median [mm]	Mittlerer Radius [mm]	Standardabweichung [mm]
Zyl. 6	7,5	7,499560	7,408533	0,422502
Zyl. 7	7,5	7,588930	7,526029	0,246862
Zyl. 8	7,5	7,453545	7,419194	0,398608
Zyl. 9	7,5	7,718365	7,686630	0,173870
Zyl. 10	7,5	7,765100	7,757408	0,128208
Zyl. 11	5,0	5,257740	5,146516	0,577249
Zyl. 12	5,0	5,143900	5,102622	0,182425
Zyl. 13	5,0	5,086810	5,067561	0,201829
Zyl. 14	5,0	4,875085	4,771043	0,566478
Zyl. 15	5,0	4,918440	4,711516	0,585470
Zyl. 20	5,0	4,915200	4,873486	0,336596
Zyl. 16	5,0	5,042520	4,906609	0,441586
Zyl. 17	5,0	4,955030	4,943652	0,306659
Zyl. 18	5,0	4,810680	4,791209	0,213731
Zyl. 19	5,0	4,897810	4,864450	0,138246
Zyl. 21	5,0	4,885380	4,771152	0,578987

Tabelle 5.25: Mittlere Radii und Standardabweichungen der halben Zylinder

Auswertung Kreisdetektion - Punkt auf Zylinderachse

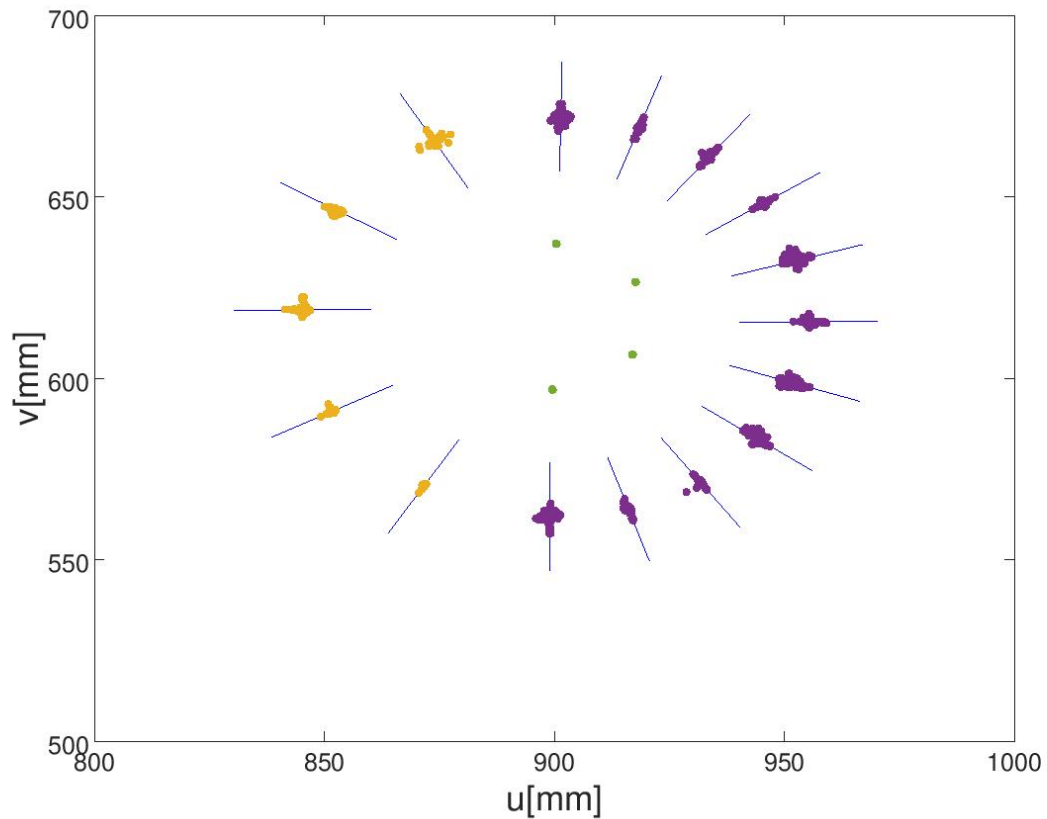


Abbildung 5.32: Darstellung der Zylinderachsenmittelpunkte projiziert auf die Modellplatte

Da der zweite sequentielle Schritt nicht nur den Radius bestimmt, sondern auch einen Punkt auf der Zylinderachse ermittelt, um die 3D Position des Zylinders im Raum zu bestimmen, wird diese im Folgenden Abschnitt ausgewertet. Der Nachteil hierbei ist, dass es wie für die Zylinderachse keine absoluten Werte für diese Punkte gibt. Zudem können für die Auswertung nicht alle Punkte entlang ihrer Zylinderachse projiziert werden, da dies eine Auswertung von 15 Ebenen verlange, wobei zwei Zylinderpaare zudem auf denselben Kreis lägen. Aus diesem Grund wird für die Auswertung nicht der ermittelte Punkte auf der Zylinderachse selbst, sondern der Mittelpunkt zwischen Start- und Endpunkt des Zylinders berechnet und verglichen. Dies ist möglich da Anfangs- und Endpunkte aus dem detektierten Kreismittelpunkt berechnet werden. Für die Berechnung werden jedoch auch die Zylinderachsen sowie die Mantelpunkte, welche für die Kreisdetektion verwendet werden, genutzt. Somit beeinflusst der Fehler des Zylinderachsenmittelpunktes auch den Fehler der Zylinderachse wieder und ist kein reiner Indikator für die Qualität des Kreismittelpunktes.

In Abbildung 5.32 werden die berechneten Zylinderachsenmittelpunkte auf die Ebene projiziert, beschrieben durch die mittlere Zylinderachse von Zylinder 2-5. Die blauen Linien, stellen die projizierten Mittelwerte der Zylinderachsen dar. Auf ihnen liegen die Zylindermittelpunkte, wobei grün die Punkte der ganzen Zylinder, lila die der Halben mit Radius 5,0mm und gelb die der Halben mit Radius 7,5mm kennzeichnet. Der Fehler der projizierten Zylindermittelpunkte wird dabei durch die Distanz zur Zylinderachse dargestellt. Hierbei sollte jedoch stets beachtet werden, dass die Zylinderachsen somit ebenfalls fehlerbehaftet sind. Je größer dieser Fehler ist, desto weniger sagt die Distanz über die eigentlichen Fehler der Achsenpunkte aus, sondern gibt nur noch einen Indikator für die Streuweite. Da jedoch die Punkte um den Achsenmittelpunkt angeordnet sind und die relativen Winkel zwischen den Achsen erhalten bleiben, wird für die Auswertung angenommen, dass der Fehler zwar vorhanden, aber klein genug ist um zumindest eine relative Aussage zu den Fehlern der Kreismittelpunkte treffen zu können.

Betrachtet man die Zylinderachsenmittelpunkte in Abbildung 5.32, so erkennt man bereits visuell sehr gut, dass die Punkte der Halbzylinder deutlich weiter gestreut sind, als die Punkte der Ganzen. Während die grünen Punkte optisch ausgewertet einem einzigen Punkt gleichen, erkennt man einige einzelne gelbe Punkte. Im Vergleich zu den Halbzylinder mit Radius 7,5mm sind die Zylinderachsenmittelpunkte von den Halbzylindern mit Radius 5mm jedoch noch einmal deutlich weiter gestreut. Trotzdem kann stets gut erkannt werden, dass sich die Punkte auf oder in nächster Nähe der jeweiligen mittleren Zylinderachse bewegen.

Zylinder	Mittelwert projizierter Achsenpunkt[mm]	minimale Distanz [mm]	maximale Distanz [mm]	Standardabweichung [mm]
Zyl. 2	[899,551061 596,935052]	0,001879	0,183149	0,016280
Zyl. 3	[900,414233 637,145877]	0,001738	0,164206	0,016312
Zyl. 4	[917,017687 606,568438]	0,000923	0,119993	0,020193
Zyl. 5	[917,632698 626,609747]	0,001434	0,125434	0,017860

Tabelle 5.26: Fehlerabstand der Zylinderachsenmittelpunkte der ganzen Zylinder

Dieses Phänomen kann ebenfalls beobachtet werden, wenn man die maximale Distanz und die Standardabweichung zwischen den ganzen Zylindern in Tabelle 5.26, mit der mittleren Distanz der halben Zylinder in Tabelle 5.27 vergleicht. Man kann dabei gut erkennen, dass die Standardabweichungen der halben Zylinder mindestens 10 Mal so groß sind, wie die der Ganzen. Da die mittlere Distanz der halben Zylinder in den meisten Fällen deutlich größer und in den wenigsten etwa gleich der maximalen Distanz der projizierten stehenden Zylinderachsenmittelpunkten ist, ist es ebenfalls eindeutig, dass halbe Kreise deutlich schlechter als Ganze detektiert werden.

Zieht man in Betracht, dass halbe Zylinder ein Grenzbereich der zu detektierenden Teilzylinder sind, ist dies akzeptable. Gerade die Positionen der Zylinder mit Radius 7,5mm sind mit einer maximalen Standardabweichung von 0,414mm und einer maximalen mittleren Distanz von 0,335mm zuverlässig und ohne große Streuung detektiert worden. Es ist also gerade bei einer Halbkreisdetektion wichtig, dass ein Zylinder möglichst viele, am besten gleichmäßig verteilte Mantelpunkte besitzt.

Zylinder	Median Distanz [mm]	Mittlere Distanz [mm]	Standardabweichung [mm]
Zyl. 6	0,139363	0,213163	0,414009
Zyl. 7	0,134753	0,194845	0,229361
Zyl. 8	0,233345	0,334971	0,367996
Zyl. 9	0,093477	0,147497	0,175013
Zyl. 10	0,101387	0,126727	0,107072
Zyl. 11	0,321996	0,483561	0,501750
Zyl. 12	0,110031	0,142013	0,172677
Zyl. 13	0,103925	0,144636	0,252326
Zyl. 14	0,177189	0,403198	0,532838
Zyl. 15	0,358673	0,498603	0,643388
Zyl. 20	0,178431	0,274297	0,313187
Zyl. 16	0,220888	0,344923	0,491740
Zyl. 17	0,108234	0,184020	0,301768
Zyl. 18	0,090324	0,150751	0,221282
Zyl. 19	0,072281	0,102173	0,130615
Zyl. 21	0,343605	0,496063	0,505122

Tabelle 5.27: Fehlerdistanz der projizierten Zylinderachsenmittelpunkte der halben Zylinder

5.3.8 Bewertung Teilzylinderdetektion auf realen Daten

Betrachtet man die Ergebnisse der realen Daten so kann festgehalten werden, dass Teilzylinder mit dem in dieser Arbeit vorgestellten Algorithmus gefunden werden können, solange nicht zu viele Ausreißer vorhanden sind und eine geringe Anzahl an Falschdetektionen nicht kritisch ist. Halbe Zylinder werden jedoch mit einer deutlich größeren Unsicherheit bestimmt und können häufiger nicht detektiert werden, insbesondere bei sehr kleinen Mantelflächen. Der Radius eines halben Zylinders wird zu dem häufiger zu klein detektiert.

Insgesamt wurden 20796 Zylinder detektiert. 19376 Zylinder davon sind gut gefunden, die restlichen 7% teilen sich in Falschdetektionen (5%) und ähnliche Zylinder, die jedoch mindestens einen falschen Parameter besitzen (2%) auf. Komplette Fehldetektionen sind sehr selten und hängen vor allem mit schneidenden Falschdetektionen zusammen.

Zusammengefasst ist es also möglich Teilzylinder bis hin zu Halbzyklindern zu detektieren, jedoch steigt die Anzahl der Falschdetektionen rasch an. Durch das Festlegen von Bedingungen, die der Kreisdetektion helfen, ist es möglich die Anzahl der Falschdetektionen zu minimieren. Falls Teilzylinder auf Realdaten mit mehr Ausreißern detektiert werden sollen, sollte die Anzahl der Punkte auf die Mantelfläche möglichst präzise gesampelt werden.

5.4 Zusammengefasste Auswertung der Zylinderdetektion

Zusammengefasst kann aus den Experimenten geschlossen werden, dass der Ansatz der Zylinderdetektion funktioniert.

Betrachtet man die Auswertung der rauschfreien Punktwolken so mussten in den Experimenten zur Auswertung der Kreisdetektion bei fehlerbehafteter Achse (5.1.2), dem simulierten Experiment der Zylinderdetektion (5.2), sowie dem simulierten Experiment zur Auswertung der Detektion von halben Zylindern (5.3) insgesamt $16000 + 21000 + 6000 = 43000$ vollständige Zylinder detektiert werden. Der Algorithmus fand insgesamt $15999 + 20895 + 4993 = 41887$, was etwa 97,4% aller Zylinder entspricht. Dabei ist es wichtig in Betracht zu ziehen, dass ein stehender Zylinder in Modell 3 zudem eine zu kleine Mantelfläche besitzt und somit aufgrund der Validierungsfunktion der Kreisdetektion in allen 1000 Versuchen nicht detektiert wurde. Somit sind insgesamt nur 113 Fehldetektionen aufgrund von ungeklärten zufälligen Fehlern zu verzeichnen. Schließt man die systematischen Fehler der 1000 Fehldetektionen aus, so wurden insgesamt 99,7% aller Zylinder detektiert.

Für die Detektion der Halbzyklinder auf rauschfreien Punktwolken waren im Abschnitt 5.3 16000 halbe Zylinder zu detektieren. Insgesamt wurden jedoch nur 15698, also etwa 98,1% aller Zylinder gefunden. Fast alle Fehldetektionen gehen auf eine Randbedingung zurück: Besitzen zwei Zylinder denselben Radius und dieselbe Achse, wobei ihre Position im Raum gleich bis auf einen Verschiebung entlang der Achse ist, so bilden die entlang ihrer Achse projizierten Zylinder den gleichen Kreis auf der Ebene. Dies kann wie sich in Abschnitt 5.3 zeigte zu Fehldetektionen führen, sollten die Mantelflächenpunkte des ersten Zylinders nicht größtenteils detektiert und gelöscht werden. Nur eine einzelne Fehldetektion stammt von einem Zylinder auf den eine solche Randbedingung nicht zutrifft.

Die Falschdetektionen für die Zylinder der simulierten Daten und der realen Daten für Modell 2 besitzen meist einen sehr kleinen Radius. Zudem weisen die falsch detektierten Zylinder eine Zylinderachse auf, die meist einen Fehler von mindestens 45° gegenüber den echten Zylinderachsen besitzen. Die falsche Zylinderachse wird im ersten Schritt der Zylinderdetektion aufgrund der Ebenenpunkte berechnet. Diese führen ebenfalls bei der Kreisdetektion dazu, dass Zylinder mit sehr kleinem Radius, die die Ebenenpunkte tangieren gefunden werden.

Um solche Falschdetektionen in der Zukunft zu vermeiden, müssen Ebenen im Vorfeld komplett aus der Punktwolke entfernt werden oder die Auswertung des Hough-Raumes für die Zylinderachsendetektion verbessert und die Validierungsfunktion der Kreisdetektion durch weitere Beschränkungen angepasst werden.

Hingegen zeigen die Falschdetektionen der realen Daten für Modell 3 eine weitere Schwäche auf. Zwar ist es möglich mit dem in dieser Arbeit vorgestellten Algorithmus Teilzylinder zu finden, jedoch müssen die Bedingungen für die Validierungsfunktion der Kreisdetektion gelockert werden, wodurch es zu einer vermehrten Anzahl an Falschdetektionen kommt. Auch ist die Detektionsrate von halben Zylindern deutlich schlechter, als die von ganzen auf realen Daten. So wurden in den 1000 Detektionen des Modells 3 15379 von 16000 halbe Zylinder detektiert (ca. 96,1%), während 3997 von 4000(6000) ganzen Zylindern detektiert wurden. Eigentlich sollten ebenfalls 6000 Vollzylinder detektiert werden, jedoch ist die Zylindermantelfläche zweier Zylinder aufgrund von Schatten zu klein. Zieht man die detektierten Zylinder des 2. Modells mit in Betracht, so wurden insgesamt $20858+3997=24855$ von $25000(27000)$, also etwa 99,4% (92,0%) aller Zylinder erfolgreich detektiert.

Insgesamt zeigt die Auswertung eindeutig, dass die in dieser Arbeit entwickelte Zylinderdetektion gegenüber der aus [22] einige deutliche Verbesserungen hinsichtlich Laufzeit und Genauigkeit liefert, wobei die Zylinderparameter keinen Beschränkungen unterliegen. Es ist somit nicht notwendig einen Radiusbereich zum Aufbauen des Akkumulators zu definieren. Gleichzeitig zeigt die Auswertung auch einige Randbedingungen auf, bei welchen der Algorithmus versagt, etwa wegen einer zu kleinen Mantelfläche oder das Ergebnis verschlechtert wird. Im folgenden Ausblick wird noch einmal auf mögliche Erweiterungen eingegangen, welche die in der Auswertung genannten Probleme minimieren können und die Zylinderdetektion noch weiter verbessern können.

Kapitel 6

Zusammenfassung und Ausblick

Zusammengefasst kann festgehalten werden, dass diese Arbeit eine randomisierte Hough-Transformation vorstellt, mit der es möglich ist Zylinder und Teilzylinder zu detektieren. Hierfür wird ein sequentieller zwei Schritt Ansatz verwendet, wobei zuerst die Zylinderachse und anschließend der Radius- sowie die Position im dreidimensionalen Raum bestimmt werden.

Aufgrund der Randomisierung ist der Algorithmus wesentlich weniger rechenintensiv als ein Standard Hough-Transformationsansatz, vorgestellt in dem als Grundlage dienenden Paper [22]. Gleichzeitig sorgt der kombinierte RHT und Least-Square Ansatz des zweiten sequentiellen Schrittes der Kreisdetektion ebenfalls für eine Verbesserung der Laufzeit, während gleichzeitig die Qualität der Kreisdetektion erhalten bleibt.

Zudem setzt der in dieser Arbeit vorgestellte Ansatz keine Anforderungen an die Zylinderparameter. Somit kann sowohl die Achse, als auch Radius und Position der Zylinder frei gewählt werden. Es ist jedoch möglich weitere Bedingungen in den vorgestellten Algorithmus einzubauen und somit seine Laufzeit und Robustheit zu verbessern. Gerade für die Detektion von Teilzylindern, insbesondere Halbzylindern, können Falschdetektionen, wenn keine Bedingungen vorhanden sind verstärkt auftreten. Aus diesem Grund ist es sinnvoll den in dieser Arbeit vorgestellten Zylinderdetektionsalgorithmus auch als einfach erweiterbare Grundlage zu betrachten, die bei Bedarf für spezifische Probleme optimiert werden kann.

Im Folgenden soll kurz ein Ausblick über mögliche sinnvolle Erweiterungen sowie Lösungen der in dieser Arbeit aufgetretenen Probleme wiedergegeben werden, sodass in Zukunft auch Zylinderdetektionen in großen Punktwolken möglich sind.

Das größte Problem des Ansatzes liegt in der großen Fehleranfälligkeit für den Fall, dass Ebenen in der Punktwolke vorhanden sind. Als Lösung wird in dieser Arbeit wie auch in [22] vorgeschlagen, vor der eigentlichen Zylinderdetektion eine Ebenendetektion durchzuführen. Dies kann wie im Falle von [15] sinnvoll sein. In dem dort vorgestellten Spezialfall konnten durch die Detektion der Decke sowie des Bodens, Bedingungen für die Zylinderachse festgelegt werden, da alle Zylinder senkrecht oder parallel zu den Ebenen liegen. Anders sieht es jedoch aus, sollten keine weiteren Bedingungen aus der Ebenendetektion gelesen werden können. In diesem Fall

ist die Ebenendetektion zwar für die Funktionalität des Algorithmus notwendig, jedoch nicht weiterführend. In einem solchen Fall ist es eigentlich sinnvoll nur sehr große Ebenen im Vorfeld herauszufiltern und dafür die Auswertung des Hough-Raumes der Zylinderdetektion stabiler zu gestalten, indem nur punktförmige Peaks und keine kreisförmigen beziehungsweise linienförmigen Maxima im Akkumulator als Zylinderachse betrachtet werden.

Zum anderen sollte die Implementierung erweitert werden, sodass die Fehldetektionen minimiert werden, die auftreten, wenn zwei Zylinder denselben Radius und dieselbe Achse besitzen und dabei nur aufgrund einer Translation entlang der Achse unterschiedlich im Raum liegen. Die einfachste Möglichkeit diesen Fall zu Umgehen ist das Einbauen eines weiteren Schwellwert, der dafür sorgt, dass nach der erfolgreichen ersten Detektion einer der zwei Zylinder nicht nur die direkt verwendeten Punkte gelöscht, sondern alle Punkte in einem gewissen Abstand zu dem Zylinder gelöscht werden. Dies wäre jedoch ein Schwellwert der sehr stark von Punktwolke zu Punktwolke variiert. Eine andere Möglichkeit ein solches Problem zu umgehen, kann über eine genauere Implementierung der Zylinderendsuche erfolgen, etwa mithilfe eines DBSCAN (Density-Based Spatial Clustering of Applications with Noise) Algorithmus welcher zuerst alle Punkte des Zylinders auf der Mantelfläche berechnet und anschließend diese komplett beziehungsweise vollständig genug löscht, sodass kein zusammenhängender Zylinder im zweiten Schritt mehr gefunden werden kann. Ein DBSCAN Ansatz könnte zudem nach der Validierung der Zylinder auf die komplette Punktwolke angewendet werden, um den systematischen Fehler der zu kurzen Zylinderlängen zu beheben.

Die oben erwähnten Erweiterungen sollen als Grundlage dienen, um die in dieser Arbeit aufgetretenen Fehler noch einmal aufzuzeigen und eine Lösung für mögliche spätere Implementierungen zu bieten. Durch die oben erwähnten Erweiterungen sollte es möglich sein auch auf größere Punktwolken mit mehr Ausreißern robust, schnell und zuverlässig Zylinder zu detektieren.

Das Ziel einer randomisierten Hough-Transformation zur Zylinderdetektion in Laserscans wurde mit dieser Arbeit erfolgreich implementiert. Sogar das Erkennen von Teilzylindern bis hin zu Halbzylindern ist möglich, solange eine geringe Anzahl an Falschdetektionen nicht kritisch ist oder weitere Bedingungen gegeben sind, um diese zu minimieren.

Anhang

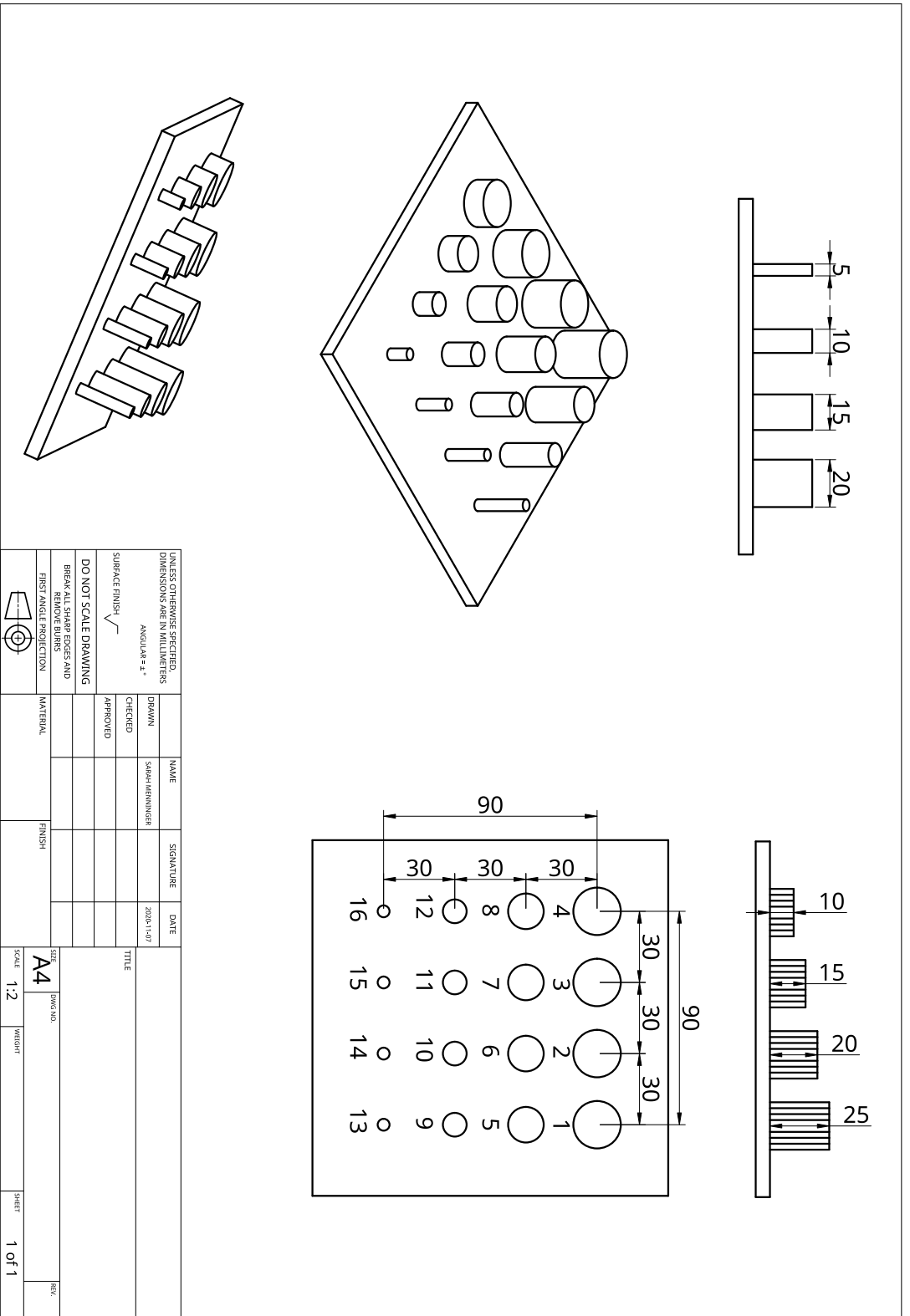


Abbildung 6.1: Darstellung des 1. Modells zur Auswertung der Kreisdetektion

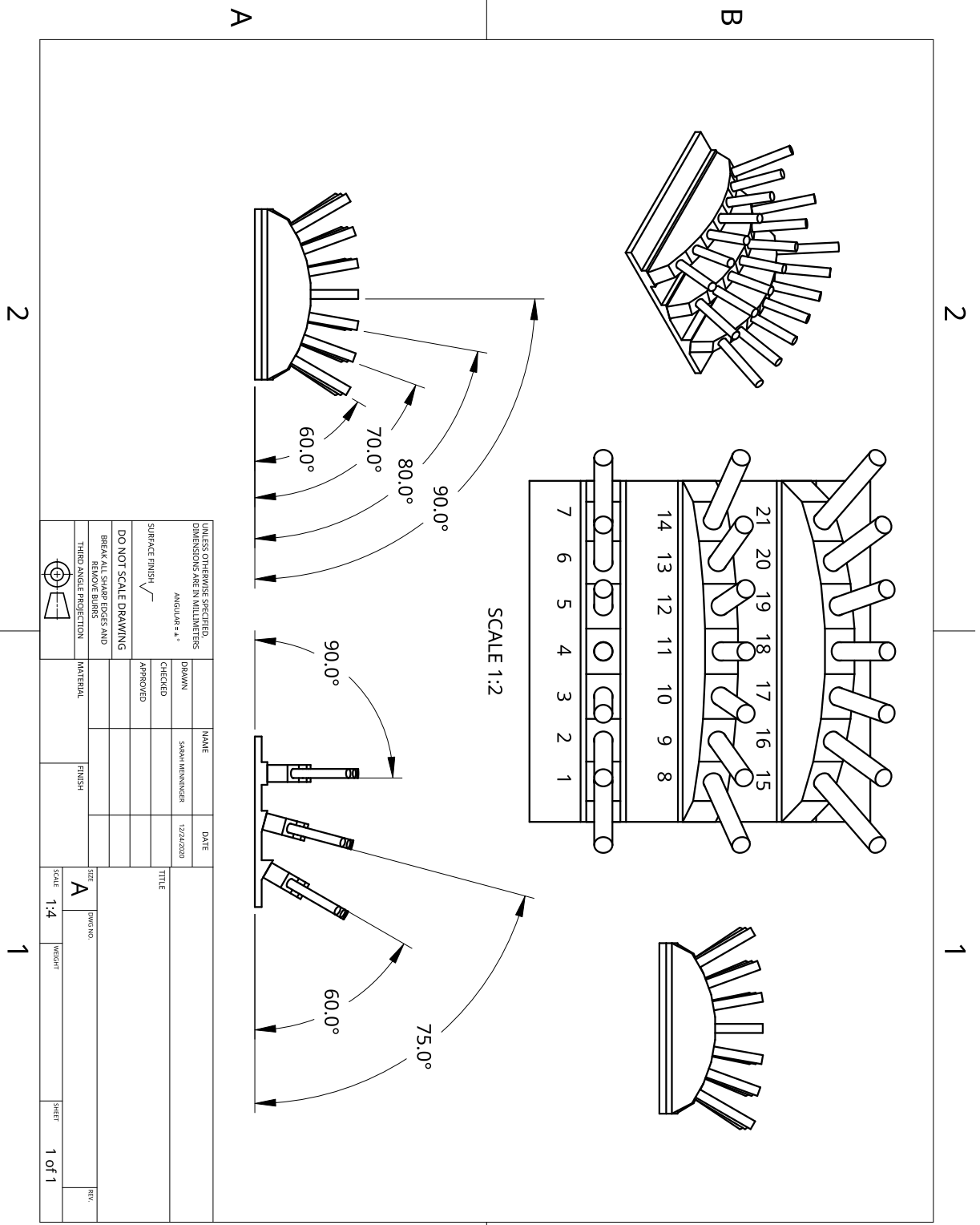


Abbildung 6.2: Darstellung des 2. Modells zur Auswertung der Detektion von Zylindern

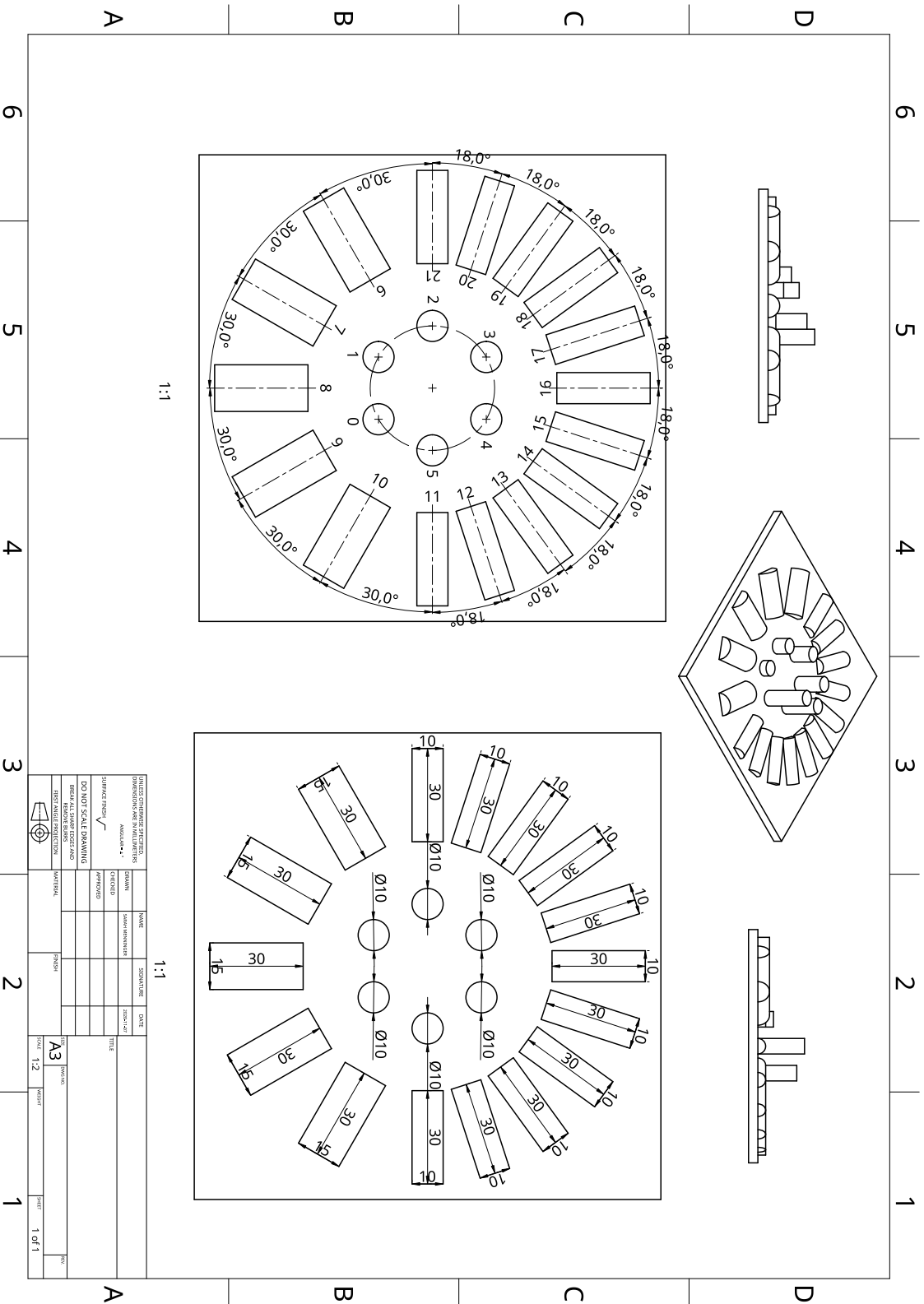


Abbildung 6.3: Darstellung des 3. Modells zur Auswertung der Detektion von halben Zylindern

Literaturverzeichnis

- [1] *3DTK - The 3D Toolkit*. <http://slam6d.sourceforge.net/>, . – Accessed: 2020-11-23
- [2] *CloudCompare*. <https://www.danielgm.net/cc/>, . – Accessed:2020-12-6
- [3] *Onshape*. <https://www.onshape.com/en/>, . – Accessed:2020-12-6
- [4] *scanCONTROL 2900-25*. https://www.micro-epsilon.de/2D_3D/laser-scanner/scanCONTROL-2900/, . – Accessed:2020-12-8
- [5] *Uni Würzburg - Rechenzentrum - 3D-Scannen*. <https://www.rz.uni-wuerzburg.de/dienste/multimedia/3d-dienste/3d-scannen/>, . – Accessed: 2020-12-2
- [6] BADINO, H.; HUBER, D. F.; PARK, Y.; KANADE, T.: Fast and accurate computation of surface normals from range images. In: *Proceedings - IEEE International Conference on Robotics and Automation* (2011), 06, S. 3084 – 3091
- [7] BORRMANN, D.; ELSEBERG, J.; LINGEMANN, K.; NÜCHTER, A.: The 3D Hough Transform for plane detection in point clouds: A review and a new accumulator design. In: *3D Research* 2 (2011), Nr. 2
- [8] CHAPERON, T.; GOULETTE, F.: Extracting Cylinders in Full 3D Data Using a Random Sampling Method and the Gaussian Image. In: *Proceedings of the Vision Modeling and Visualization Conference* (2001), 01, S. 35–42
- [9] GOLUB, G. H.; LOAN, C. F. V.: *Matrix Computations*. Bd. 4. Baltimore, Maryland 21218-4363 : The Johns Hopkins University Press, 2013
- [10] GUO, S.; ZHANG, X.; ZHANG, F.: Adaptive Randomized Hough Transform for Circle Detection using Moving Window. In: *2006 International Conference on Machine Learning and Cybernetics*, 2006
- [11] In: HENNING, B.; BESSLICH, P. W.: *Hough-Transformation*. Berlin, Heidelberg : Springer, 1991, S. 101–121

- [12] HOPPE, H.; DEROSE, T.; DUCHAMP, T.; McDONALD, J.; STUETZLE, W.: Surface Reconstruction from Unorganized Points. In: *Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques*. New York, USA : Association for Computing Machinery, 1992
- [13] KIMME, C.; BALLARD, D.; SKLANSKY, J.: Finding Circles by an Array of Accumulators. In: *Communications of the ACM* 18 (1975), Nr. 2, S. 120 – 122
- [14] KIRYATI, N.; ELДАР, Y.; BRUCKSTEIN, A.M.: A probabilistic Hough transform. In: *Pattern Recognition* 24 (1991), Nr. 4, S. 303 – 316
- [15] LIU, Y.; ZHANG, J.; HOU, J.; REN, J.; TANG, W.: Cylinder Detection in Large-Scale Point Cloud of Pipeline Plant. In: *IEEE Transactions on Visualization and Computer Graphics* 19 (2013), Nr. 10, S. 1700–1707
- [16] LUKÁCS, G.; MARTIN, R.; MARSHALL, D.: Faithful Least-Squares Fitting of Spheres, Cylinders, Cones and Tori for Reliable Segmentation. In: *Proceedings of the 5th European conference on computer vision*, 1998
- [17] LUTTON, E.; MAITRE, H.; LOPEZ-KRAHE, J.: Contribution to the determination of vanishing points using Hough transform. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16 (1994), Nr. 4, S. 430–438
- [18] In: MARSCHNER, S.; SHIRLEY, P.: *Fundamentals of Computer Graphics*. 3rd. USA : A. K. Peters, Ltd., 2009, S. 25–30
- [19] NURUNNABI, A.; SADAHIRO, Y.; LINDENBERGH, R.: ROBUST CYLINDER FITTING IN THREE-DIMENSIONAL POINT CLOUD DATA. In: *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-1/W1* (2017), 05, S. 63–70
- [20] PAPULA, L.: *Mathematische Formelsammlung - Für Ingenieure und Naturwissenschaftler*. Bd. 9. Wiesbaden : Viewegs Fachbücher der Technik, 2006
- [21] PRESS, W. H.; TEUKOLSKY, S.A.; VETTERLING, W. T.; FLANNERY, B. P.: *Numerical recipes - The Art of Scientific Computing*. Bd. 3. The Edinburgh Building, Cambridge CB2 8RU : Cambridge University Press, 2007
- [22] RABBANI, T.; HEUVEL, F.: Efficient Hough transform for automatic detection of cylinders in point clouds. In: *Proc ISPRS Workshop Laser Scan 2005, ISPRS Arch* 36 (2005), 01
- [23] SCHNABEL, R.; WAHL, R.; KLEIN, R.: Efficient RANSAC for Point-Cloud Shape Detection. In: *Computer Graphics Forum* 26 (2007), 06, Nr. 2, S. 214–226

- [24] TAUBIN, G.: Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. In: *IEEE Transaction on Pattern Analysis and Machine Intelligence* 13 (1991), 11, Nr. 11, S. 1115–1138

- [25] THOMAS, S.M.; CHAN, Y.T.: A simple approach for the estimation of circular arc center and its radius. In: *Computer Vision, Graphics, and Image Processing* 45 (1989), Nr. 3, S. 362 – 370

Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

A handwritten signature in black ink, appearing to read 'S. Meninger'. The signature is fluid and cursive, with a long horizontal stroke extending to the right.

Würzburg, Januar 2021