

Bachelorarbeit

Vergleich von Konfigurationen aktiver Marker zur Erkennung und Schätzung ihrer Pose in trübem Gewässer

Johanna Tholen
tholen@uni-bremen.de
Matrikelnummer: 4444452

02.10.2021

Erstgutachter	Prof. Dr. Frank Kirchner
Zweitgutachter	Prof. Dr.-Ing Udo Frese
Betreuer	Dr.-Ing. Marc Hildebrandt
	Dipl.-Inf. Sascha Arnold

Fachbereich 3 - Informatik
Universität Bremen

Erklärung

Ich versichere, die Bachelorarbeit ohne fremde Hilfe angefertigt zu haben. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Alle Stellen, die wörtlich oder sinngemäß aus Veröffentlichungen entnommen sind, sind als solche kenntlich gemacht.

Bremen, den 2. Oktober 2021

Johanna Tholen

Inhaltsverzeichnis

1 Einleitung	3
1.1 Einleitung	3
1.2 Zielsetzung	4
1.3 Aufbau der Arbeit	4
2 Stand der Technik	5
2.1 Verwandte Arbeiten	5
2.1.1 Erkennung von Lichtquellen auf astronomischen Bildern	5
2.1.2 Posenschätzung unter Nutzung aktiver Marker	6
2.2 Marker zur Posenbestimmung	7
2.3 Detektion	7
2.4 Posenschätzung	8
3 Implementierung	9
3.1 Detektor	10
3.1.1 Kandidatensuche	10
3.1.2 Suche des Mittelpunkts	11
3.1.3 Bestimmung der Größe	12
3.1.4 Entscheidung ob es sich um ein Licht handelt	13
3.2 Identifizieren	13
3.2.1 Zuordnung	13
3.2.2 An- und Auszeiten analysieren	13
3.2.3 Zugehörigen echten Marker finden	14
3.3 Posenschätzung	14
4 Hardware	16
4.1 Marker-Konfigurationen	16
4.1.1 Welches Merkmal zum Identifizieren	16
4.1.2 Auswahl der Blinkfrequenzen	17
4.1.3 Andere Merkmale	17
4.2 Technische Details der Lichter	19
4.3 Technische Details der Kamera	19
4.4 Versuchsaufbau	20
4.4.1 Versuch	20
5 Evaluation	23

5.1	Detektor	23
5.1.1	Erkennung in klarem Wasser	23
5.1.2	Erkennung bei Trübung	24
5.1.3	Entscheidung ob es sich um ein Licht handelt	24
5.2	Identifizierer	26
5.3	Posenschätzer	26
5.4	Unterschiede zwischen Markerkonfigurationen	28
5.4.1	Binkfrequenzen	33
6	Fazit	34
Anhang		36
Literaturverzeichnis		38

Kapitel 0:

1 Einleitung

1.1 Einleitung

Für die Erkundung von tiefen Gewässern gewannen Autonome Unterwasser Fahrzeuge (AUV's) durch die Forschung an autonomen Systemen in den letzten Jahrzehnten zunehmende Bedeutung. Sie werden zu Standardwerkzeugen in unterschiedlichen Anwendungsbereichen wie Umweltvermessung, Geologie, Archäologie, Kabelinspektion und verschiedenen anderen Bereichen der Industrie und des Militärs. [9] Wegen der beschränkten Akkukapazität werden bei Nutzung von AUVs oft Dockingstations verwendet, an denen es seinen Akku laden kann und gesammelte Daten überträgt. Um dieses Konzept umzusetzen, muss das AUV in der Lage sein, autonom an eine solche Dockingstation anzudocken.

In dem EurEx-Luna Projekt sollen Methoden erforscht werden mit denen der Jupiter-Mond Europa untersucht werden kann. Forscher vermuten auf Europa einen flüssigen Salzwasser-Ozean. [4] Dieser ca. 100 Kilometer tiefe Ozean soll sich unter einer 3-15 Kilometer dicken Eisschicht befinden. Dieser Ozean ist laut Forschern einer der wahrscheinlichsten Orte im Sonnensystem für außerirdisches Leben. [11] Im EurEx-Luna Projekt wird untersucht, wie sich der unter dem Eis vermutete Ozean mit Hilfe eines AUVs erforschen lässt. Dieses AUV müsste in der Lage sein an einer Dockingstation anzudocken, um den Akku aufladen zu können und eine Datenübertragung zur Erde zu ermöglichen.

Um in die Nähe der Dockingstation zu finden, werden akustische Signale verwendet. Für eine zuverlässigere Erkennung der Dockingstation, sollen visuelle Daten verwendet werden. Aus diesem Grund soll die Posenschätzung in der Nähe der Dockingstation durch die Auswertung eines Kamerabildes unterstützt werden. Dabei sollen aktive Marker, worunter man eine Anordnung von Lichtquellen versteht, genutzt werden. Ihr Vorteil besteht darin auch in trübem Gewässer und unter verschiedenen Lichtverhältnissen gut erkennbar zu sein. [14]

Für die robuste Bestimmung der Pose der Dockingstation ist die Konfiguration der Marker von Bedeutung. Wobei die Konfiguration die Anzahl, die Anordnung, die Blinkfrequenz und Farbe berücksichtigen kann. Um die Marker voneinander zu unterscheiden kann z.B. die Blinkfrequenz der Marker unterschiedlich gewählt werden. So kann man durch Auswertung mehrerer Bilder die Marker zuordnen. [3] In einem anderen Ansatz wurden zur Unterscheidung der Marker Farben genutzt. [14] In dieser Arbeit soll untersucht werden, welche Konfiguration sich eignet, um daraus die relative Pose der Dockingstation robust

zu bestimmen. Dabei sollen verschiedene Konfigurationen hinsichtlich der Erkennbarkeit der Marker untersucht werden. Es wird insbesondere evaluiert, wie genau die Erkennung ist, wie viele Ausreißer es gibt und in wie vielen Bildern sie möglich ist.

1.2 Zielsetzung

In der Arbeit sollen verschiedene sinnvolle Konfigurationen von Markern erarbeitet werden, die im späteren Verlauf evaluiert werden können. Außerdem soll eine Erkennung der Marker implementiert werden, welche auch unter schlechten oder nicht optimalen Sichtverhältnissen noch Marker erkennen kann. Von der Erkennung ausgehend wird auch eine Schätzung der Pose implementiert werden. Anschließend wird evaluiert, bei welcher Konfiguration die Erkennung der Marker unter Einbeziehung unterschiedlicher Parameter am robustesten gelingt. Die Parameter sind: die Trübung des Wassers, die Distanz der Marker und die Lichtbedingungen. Um die sinnvollsten Markerkonfigurationen zu finden, soll im Wesentlichen folgendermaßen vorgegangen werden:

1. Erarbeitung möglicher Konfigurationen und Festlegung auf die geeigneten Kandidaten für die genaueren Untersuchungen und den Vergleich
2. Erkennen der Marker im Kamerabild
3. Ableiten der Pose
4. Testen der verschiedenen Konfigurationen in Experimenten

1.3 Aufbau der Arbeit

Zunächst wird im zweiten Kapitel in die Grundlagen und den Stand der Technik eingeführt. Darauf wird im dritten Kapitel erklärt wie vorgegangen wurde, um das vorgestellte Problem zu untersuchen. Es wird zunächst die Erkennung der Marker, dann die Identifizierung und die Schätzung der Pose erklärt.

Im vierten Kapitel werden die Marker Konfigurationen und der Versuchsaufbau vorgestellt. Anschließend folgt die Evaluation, in der die Versuche ausgewertet werden und anhand von ihnen die Software bewertet wird. Danach wird noch ein Fazit geschlossen.

2 Stand der Technik

2.1 Verwandte Arbeiten

Erkennung von Lichtquellen auf astronomischen Bildern

Marc Masias Moyset aus 2017 [15] hat in seiner Doktorarbeit mehrere Methoden zur Erkennung von Lichtquellen im All untersucht. Lichtquellen im All sehen denen im Wasser recht ähnlich, daher ist dies auch für uns interessant. Eine dieser Methoden basiert auf der von ihm definierten Radial Contrast Function. Für die Erkennung wird das Verhalten von Licht aus einer Punktquelle zu Nutze gemacht. Nach dieser Annahme sind die Bildpunkte in Richtung der Lichtquelle am hellsten und nehmen kreisförmig um diesen Punkt ab. Mithilfe der Radial Contrast Function kann das Bild auf ebenso ein Phänomen untersucht werden. Dazu wurde zunächst die Radial Contrast Function für den Pixel (i,j) wie folgt definiert:

$$RcF(R) = \left(I(i,j) - \frac{\sum I(k,l)(R)}{N_{neigh}(R)} \right) / I(i,j) \quad (2.1)$$

R steht für die Größe des Radius, für den die Funktion berechnet werden soll. Die

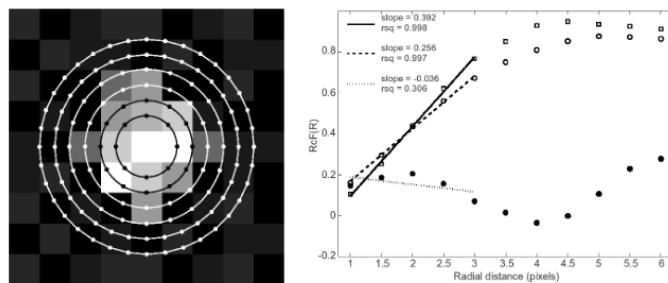


Abbildung 2.1: Grafik aus: [15]

Funktion $I()$ steht für die Helligkeit des Pixels. k nimmt alle Werte von $i - R$ bis $i + R$ an und l alle Werte von $l + R$ bis $l - R$. Es wird also immer der Kontrast zu der Helligkeit der Mitte gebildet. Je niedriger die Helligkeit des Umkreises wird, desto größer wird der

Kontrast. Man erwartet also, dass je größer R gewählt wird, das Ergebnis der R_cF (Radial Contrast Function) auch größer wird.

Für die Suche nach Lichtern, wird jeder Kandidatenpunkt analysiert. Hierfür wird die R_cF für den Pixel und immer größere Umkreise des Pixels berechnet. Wenn es sich um ein Licht handelt, sollte der Radial Contrast je größer der Umfang ist immer größer werden. Diese Steigung sollte außerdem linear verlaufen. Um jetzt zu analysieren ob es sich um ein Licht handelt, wird untersucht, ob es sich um eine lineare Funktion mit einer Mindeststeigung handelt. Hierfür wird anhand der Werte eine lineare Funktion angenähert und einmal die Steigung bestimmt, die einen Mindestwert aufweisen muss. Um die Güte der Annäherung zu bestimmen, wird der Determinationskoeffizient berechnet und es wird erwartet, dass er höher ist als der definierte Mindestwert. Anhand dieser Werte wird entschieden, ob der Pixel ein möglicher Kandidat sein könnte. Wenn in einem gewissen Umkreis mehrere Pixel, mit guten Werten vorkommen, wird davon ausgegangen ein Licht gefunden zu haben.

Posenschätzung unter Nutzung aktiver Marker

Josep Bosch et al. (2016) [3] haben ein System entwickelt mit dem AUVs, die in einer Formation fahren, Posen der anderen AUVs mithilfe von aktiven Markern schätzen und sich gegenseitig verfolgen können. Bei der Erkennung der Lichter wurden zunächst Kandidaten gesucht. Hierfür wurde das Kantenbild mithilfe des Sobel Filters bestimmt und die hellsten Punkte als Kandidaten für ein Licht gewählt. Hierbei war Voraussetzung, dass die Lichter eine Mindestentfernung zueinander haben.

Für diese Kandidaten wurde daraufhin der Mittelpunkt gesucht, indem sich nur ein kleiner Ausschnitt um den Kandidatenpunkt angesehen wird und dann das Lokale Maximum in der Nähe der Mitte des Bildes gesucht wird.

Daraufhin wird überprüft, ob der Kandidat eine realistische Helligkeit und Größe hat. Die Größe wird mit einem zur Radial Contrast Function ähnlichen Methode bestimmt. Es wird in der Mitte des Kandidaten angefangen und in immer größeren Radien der Helligkeitsdurchschnitt bestimmt. Wenn die Helligkeiten der Nachbarn nicht mehr dunkler werden, kann man davon ausgehen, dass der Lichtkegel des Markers verlassen wurde und den momentanen Radius als Radius des Lichtkegels betrachten.

Sascha Flögel et al. (2018) [7] haben auch einen Ansatz zur Posenschätzung unter Nutzung von aktiven Markern erstellt und getestet. Die aktiven Marker wurden in einem unsymmetrischen Kreuz mit jeweils vier kollinearen Markern zusammengestellt. Zu der Identifikation kann das Kreuzverhältnis der 4 Marker dienen, weil es sich unter perspektivischer Projektion invariant verhält. Zur Posenschätzung wurde ein Algorithmus genutzt, der das Perspective-n-Point Problem löst. Das Perspective-n-Point Problem beschreibt die Berechnung der Pose ausgehend von bekannten intrinsischen Kalibrierungsparametern der Kamera, sowie der räumlichen 3D-Position und 2D-Bildposition von n bekannten Punkten.

2.2 Marker zur Posenbestimmung

Es gibt verschiedene Standards und Konzepte von Markern die für die Posenschätzung oder zur Identifizierung von Objekten genutzt werden können. Die bekanntesten Konzepte vereint dabei eine Schwarz-Weiße Farbwahl mit aufgedruckten Quadranten. Der hohe Helligkeitskontrast ermöglicht eine gute Auswertung durch Algorithmen der Bildverarbeitung. Die Ecken der einzelnen Quadrate können erkannt werden und so aus einem Marker eine Vielzahl von Punkten erkannt werden. Aus der bekannten Geometrie der Marker, und ihrem Auftreten auf dem Bild, kann sowohl Position und Orientierung, also die vollständige Pose, aus einem Marker berechnet werden. Dadurch das auch erkannt wird wie weiße und schwarze Quadrate angeordnet sind, können diese Quadrate als bits interpretiert werden. Durch unterschiedliche Anordnungen der Quadrate ergeben sich verschiedene bit-Sequenzen. So können die Marker voneinander unterschieden werden. Ein solcher Marker ist der Apriltag Marker. [16].

In dieser Arbeit wurden aktive Marker genutzt. Aktiv heißt in diesem Zusammenhang, dass sie von sich aus leuchten. Das ist aufgrund der Unterwasserdomäne von Vorteil. Der Marker muss auch in trübem und dunklem Wasser gut erkannt werden, dies gelingt besser wenn der Marker leuchtet.

2.3 Detektion

Bildverarbeitung ist ein zentrales Thema in der Informatik. Eines der entscheidenden Probleme dieses Bereichs ist die Objektdetektion. Je nach gesuchtem Objekt sind die Methoden zahlreich und auch sehr unterschiedlich[1]. Komplexe Objekte erfordern dabei auch komplexere Methoden, oft auch lernende Verfahren, da die Objekteigenschaften nicht mehr analytisch zu beschreiben sind. Für einfache, eher geometrische Objekte kann man jedoch mathematisch definierbare Eigenschaften beschreiben. Solche Objekte lassen sich mit, für den Stand der Technik recht simplen Verfahren, sehr genau und robust detektieren. Die genaue Bildposition des Objekts ist dabei nicht immer von Bedeutung. Für die Anwendung dieser Arbeit ist sie aber von zentraler Bedeutung. Neben der Komplexität des eigentlichen Objekts erschwert der Umstand, dass dieselben Objekte in verschiedenen Situationen sehr unterschiedlich aussehen können. Es gibt viele Faktoren, die dabei eine Rolle spielen. Zu diesen zählen zum Beispiel Verdeckung, verschiedene Helligkeit und Reflektionen. In unserem Fall sehen die Lichter je nachdem wie hell sie sind oder wie nah wir an ihnen dran sind anders aus. Ein Ziel der Detektion ist also mit solchen Faktoren gut umgehen zu können.

Wenn man analytisch vorgeht, kann man sich an gewissen Merkmalen, die das Objekt auszeichnen (bei den Lichtern zum Beispiel die Helligkeit) orientieren. Anhand dieser Merkmale probiert man die Objekte zu identifizieren. Bei Recht vielschichtigen Objekten wie einem Tier oder Menschen kommt solch ein Ansatz allerdings an seine Grenzen, weil die Erkennung eines Menschen sehr viele verschiedene Merkmale hat, die zusammen wirken und nur durch das Gesamtbild erkannt werden. Analytische Verfahren eignen

sich für kontrollierbare Umgebungen, wo man davon ausgehen kann, dass die Objekte nicht stark unterschiedlich wirken. Außerdem ist es vorteilhaft, wenn die zu erkennenden Objekte klare Merkmale haben und nicht zu facettenreich sind.

DeepLearning kann mit unterschiedlichen Umgebungen und facettenreichen Objekten gut umgehen. Um aber etwas lernen zu können muss es möglich sein viele Trainingsdaten zu haben, oder noch machen zu können, die auch alle Grenzfälle abdecken. Ein Problem an DeepLearning ist, dass man nicht kontrollieren kann an was sich das Netz orientiert und man es so nicht so einfach umpolen kann.

2.4 Posenschätzung

Posenschätzung ist ein wichtiges Gebiet in der Robotik, da zum Beispiel für die Navigation entscheidend ist, wo sich der Roboter befindet. Hierfür gibt es verschiedene Methoden. In dieser Arbeit wird sich auf die Schätzung der Pose durch visuelle Marker beschränkt.

Ziel der Posenschätzung ist es mithilfe der Kamera, die Orientierung der Kamera in Bezug auf ein Objekt zu bestimmen. Die Pose ist dabei die Kombination aus Position und Orientierung des betrachteten Körpers.[\[18\]](#) Gegeben erkannter Bildpunkte von detektierten Objekten im Bild und bekannten Positionen der detektierten Objekte lässt sich ein Bezug herstellen, durch den die Berechnung der Pose möglich ist, die die erkannten Bildpunkte zu den dreidimensionalen Position der zugehörigen erklärt. Je nach Anzahl der Objekte lässt sich die Pose nämlich eindeutig bestimmen. Die Ausnutzung dieser Bezüge zur mathematischen, analytischen Bestimmung der Pose ist ausführlich untersucht und erforscht[\[13\]\[20\]\[5\]](#). Aufgrund von Messunsicherheit und Rauschen in der algorithmischen Detektion als auch auf sensorischer Ebene sind diese Verfahren oft ungenau[\[8\]](#). In robotischen Anwendungen werden deshalb vor allem numerische Verfahren genutzt. Alle diese Verfahren vereint, dass ein Modell der Kamera benötigt wird, das vorhersagt, wo ein Objekt im Bild zu sehen sein müsste, gegeben seiner Position relativ zur Kamera. Es gibt viele mögliche Parameter, die in diesem Modell berücksichtigt werden könnten, je nach steigender Komplexität kann die Schätzung dadurch etwas verbessert werden[\[6\]](#). Da die Position der Objekte bekannt ist, ergibt sich die relative Position nur aus der Pose der Kamera. Die erwarteten Bildpunkte lassen sich aus der relativen Position und des Modells der Kamera berechnen. Wenn die Bildpunkte den gemessenen übereinstimmen, muss die geschätzte Pose der Kamera, aus denen die erwarteten Bildpunkte berechnet wurden, ungefähr der Pose entsprechen, in denen die tatsächlichen Bildpunkte aufgenommen wurden. Es liegt also ein direkter Bezug von der geschätzten Pose zu den Bildpunkten der sich zur Schätzung ausnutzen lässt. Da wir nicht sofort die richtige Pose kennen, wird man einen Fehler zwischen errechneten Bildpunkten, also jenen die aus dem Kameramodell und der relativen Pose berechnet wurden, und den gemessenen Bildpunkten beobachten. Dieser Fehler ist erst mal in Form einer Distanz auf der Bildebene definiert. Die Schätzung der Pose kann somit als Optimierungsproblem für eben diesen Fehler und der Pose als Entscheidungsvariable definiert werden.

3 Implementierung

Das Ziel der Implementierung ist, dass aus dem Kamerabild der Marker eine Pose abgeleitet wird. Die Software wurde in verschiedene Komponenten unterteilt. Einmal in den Detektor, der das Bild als Eingabe bekommt und in ihm die Lichter erkennt und ausgibt. Das ausgegebene geht als Eingabe zum Identifizierer, der die Lichter den Lichern aus den vorherigen Bildern zuordnet, die Blinkfrequenzen erkennt und identifiziert. Die identifizierten Lichter gehen dann an den Posenschätzer weiter, der aus den identifizierten Lichtern die Pose schätzt und diese ausgibt.

Die Software wurde in dem Framework ROS (Robot Operating System) implementiert. In ROS kann man Nodes erstellen, die jeweils ein eigener Prozess sind, der parallel mit anderen Nodes laufen kann. Der Detektor, Identifizierer und Posenschätzer wurden jeweils als Node angelegt, wobei die grundlegenden Funktionen als Bibliothek angelegt wurden. Wenn die Nodes laufen können sie Nachrichten verschicken und empfangen. Das Bild, die erkannten und identifizierten Lichter und die Pose werden also als Nachricht definiert. Wenn der Detektor die Lichter erkannt hat, veröffentlicht er die erkannten Lichter als Nachricht und der parallel laufende Identifizierer reagiert auf sie und veröffentlicht selber die nächste Nachricht.

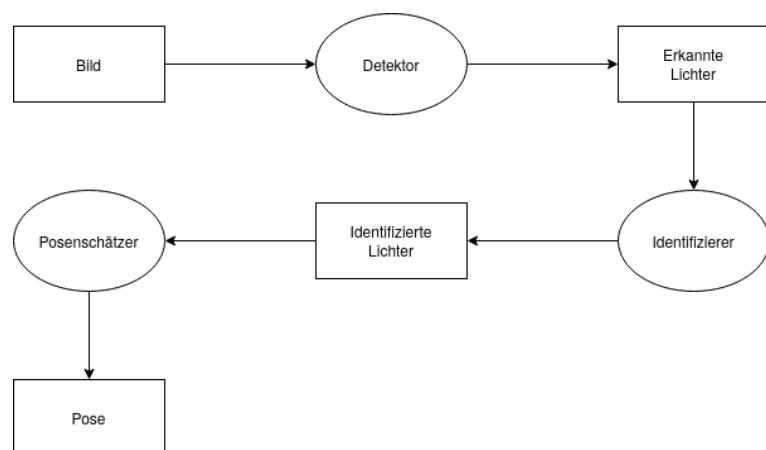


Abbildung 3.1: Software-Architektur

3.1 Detektor

Der Detektor muss mit unterschiedlichen Größen und Formen zurechtkommen und zuverlässig und genau den Mittelpunkt des Lichts und dessen Größe bestimmen können. Es wurde sich für ein analytisches Vorgehen entschieden, weil für ein lernendes Verfahren nicht genug Trainingsdaten zur Verfügung gestanden hätten. Außerdem sind Lichter nicht so facettenreich und man kann sie auch gut analytisch erkennen.

Der Algorithmus lässt sich schematisch in folgende Punkte einteilen:

1. Kandidatensuche
2. Suche des Mittelpunkts
3. Bestimmung der Größe
4. Bestimmung, ob es sich tatsächlich um ein Licht handelt

Der Detektor nimmt als Eingabe ein Schwarz-Weiß Bild und gibt die erkannten Lichter mit ihren Attributen aus, sodass sie vom Identifizierer weiterverarbeitet werden können.

Kandidatensuche

Das Verfahren kann aufgrund der Laufzeit nicht für jeden Pixel des Bilds durchlaufen werden. Außerdem lassen sich die meisten Pixel als mögliche Kandidaten leicht ausschließen. Eben dazu dient die Kandidatensuche: Durch ein Verfahren mit deutlich geringerer Laufzeit wird eine Vorauswahl getroffen welche Pixel überhaupt als mögliche Lichter in Frage kommen. Die Zahl der Kandidaten wurde hier auf 20 begrenzt. Um Kandidaten zu finden ist es wichtig die Eigenschaften der Lichter zu betrachten. Bei Lichtern fällt auf, dass sie meistens einen recht starken Helligkeitsunterschied zu ihrer Umwelt haben. Dies Unterscheidet sich im Gegensatz zu großen Flächen, die beispielsweise durch einen Scheinwerfer angeleuchtet werden. Zwar großflächig hell sind, aber eben nie einen großen Helligkeitsunterschied zu ihrer direkten Umgebung haben.

Um diesen Effekt zu Nutzen wurde wie in [3] ein Sobelfilter auf das Bild gelegt, um Kanten zu erkennen. Hieraus werden jetzt die x-hellsten Punkte gewählt. Der Sobelfilter ist wie folgt definiert:

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} * A \quad (3.1)$$

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * A \quad (3.2)$$

$$G = \sqrt{G_x^2 + G_y^2} \quad (3.3)$$

Es gibt aber auch immer Lichter, die nicht so klare Helligkeitsunterschiede mit dem direkten Umfeld aufweisen, zum Beispiel weil sie recht groß sind. Deswegen werden außerdem noch die Hälfte der Kandidaten aus dem Originalbild gewählt.

Wir erwarten allerdings auch einen gewissen Abstand zwischen den Lichtern, sodass kein Licht doppelt als Kandidat gewählt werden kann.

Suche des Mittelpunkts

Im nächsten Schritt ist es wichtig den Mittelpunkt zu finden. Je genauer dieser gefunden wird, desto genauer kann später die Pose geschätzt werden.

Es gibt verschiedene Möglichkeiten einen Mittelpunkt zu finden. Die intuitivste wäre einfach den hellsten Punkt zu nehmen. Das ist allerdings sehr fehleranfällig und in diesem Fall auch schwer verwirklichbar, denn bei den meisten Lichtern erreichen viele Punkte die maximale Helligkeit.

Deshalb wird iterativ vorgegangen:

Folgende Schritte werden unternommen:

1. Es wird ein kleiner Bildausschnitt um den Kandidatenpunkt kreiert.
2. Es soll eine Helligkeitsverteilung gewonnen werden. Hierfür wird der Ausschnitt auf die Pixel, die heller als der Helligkeits-Durchschnitt aller Pixel ist gefiltert.(Gleichung 3.4)¹
3. Aus den gefilterten Pixeln, wird jetzt der Mittelpunkt berechnet.
4. Es wird überprüft, ob der gefundene Mittelpunkt in unmittelbarer Nähe zum Kandidaten ist. Wenn ja, geht es mit Punkt 6 weiter, wenn nein mit Punkt 5.
5. Es wird wieder mit Punkt 1 begonnen. Der Bildausschnitt wird nun aber größer gewählt und der gefundene Mittelpunkt ist der neue Kandidatenpunkt.
6. Der Mittelpunkt ist gefunden und wird zurückgegeben.

Im folgenden Bild kann man ein Beispiel der Suche sehen. Der rote Punkt ist immer der Kandidat und der blaue der berechnete Mittelpunkt. Man sieht wie das Bild immer größer gewählt wird und sich der berechnete Mittelpunkt immer mehr der echten Mitte annähert.

$$dst(x, y) = \begin{cases} \text{maxValue} & src(x, y) > T(x, y) \\ 0 & \text{otherwise} \end{cases} \quad (3.4)$$

¹<https://opencv.org/>

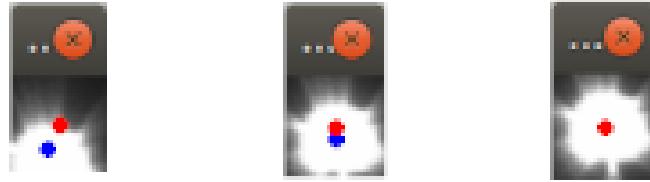


Abbildung 3.2: Suche nach Mittelpunkt

Bestimmung der Größe

Bestimmung der Größe unter Nutzung der Radial Contrast Function

Bei der Schätzung der Größe wird sich an der Radial Contrast Function [15] und ihrer Nutzung in [3] orientiert. Im Paper wurde ein recht ähnlicher Ansatz wie die Radial Contrast Function gewählt. Es wurde wie auch bei der Radial Contrast Function erwartet, dass die Lichter nach außen hin dunkler werden. Es wird so vorgegangen, dass man die Helligkeit der Nachbarkreise iterativ analysiert. Hierbei wird erwartet, dass die Kreise immer dunkler werden. Wenn sie irgendwann bei einer Helligkeit bleiben, weiß man, dass die Größe gefunden wurde.

Dieser Algorithmus wurde genutzt, um die Größe zu bestimmen. Die Lichter, die wir aufgenommen haben, waren auf den Bildern aber meist sehr groß, sodass sehr viele Pixel die hellste Helligkeit hatten und man in den ersten Schritten keine Veränderung der Durchschnittshelligkeit bemerkte. Um damit umzugehen, wurde der Algorithmus abgeändert. Solange ein Umkreis nicht signifikant dunkler war als die Mitte, werden die Umkreise nicht berücksichtigt. Es wird erst gestartet, wenn einmal ein Umkreis signifikant dunkler war. Es werden die überbelichteten Pixel also ausgeschlossen.

Im folgenden Bild sind alle Pixel von denen die Durchschnittshelligkeit berechnet wird blau markiert.



Abbildung 3.3: Ablauf bei der Größenerkennung

Entscheidung ob es sich um ein Licht handelt

Nicht jeder Kandidat ist wirklich ein Licht, deswegen müssen Kandidaten nach verschiedenen Kriterien aussortiert werden, um weniger Falsch Positive zu erhalten. Um solche Falscherkennungen zu vermeiden, gibt es eine Mindesthelligkeit der Lichter. Außerdem gibt es eine Maximal- und Minimalgröße. Beim Durchlaufen des Algorithmus der auf die Größe prüft, wird auch geprüft ob die Kandidaten die richtige Form haben. Wenn sie nicht annähernd die richtige Form haben, kann die Größe nämlich nicht bestimmt werden und der Kandidat wird aussortiert.

Die Form wird auch noch durch die Nutzung der Radial Contrast Function überprüft. Es wird, wie in Unterabschnitt 2.1.1 beschrieben, der Algorithmus, auf den Mittleren Pixel des Lichts angewendet und die Steigung und die Güte der Annäherung werden bewertet. Wenn die beiden Werte besser als der Schwellwert sind und auch die vorherigen Kriterien erfüllt sind, gilt der Kandidat als erkanntes Licht.

3.2 Identifizieren

Der Identifizierer arbeitet mit den erkannten Lichtern des Detektors, er erkennt die Blinkfrequenzen. Da einem Marker immer eine gewisse Blinkfrequenz zugeordnet ist, können die erkannten Markerfrequenzen mit den echten Markerfrequenzen verglichen werden. Anhand dessen kann entschieden werden, ob das erkannte Licht tatsächlich zu einem echten Marker gehört.

Zuordnung

Da wir von einer statischen Umgebung ausgehen, kann die Zuordnung recht einfach gehalten werden. Es werden die im neuen Bild erkannten Lichter mit den vorher erkannten, in den Attributen Position und Helligkeit abgeglichen und anhand dessen identifiziert. Die Größe konnte nicht zur Zuordnung verwendet werden, weil die Marker beim an- und ausgehen ihre Größe stark verändern und dadurch manchmal nicht richtig zugeordnet werden könnten.

Sobald ein Marker länger als $1.5 \cdot$ maximale Zeit der Auszeit nicht gesehen wurde, wird er aus der Liste der bereits erkannten Marker entfernt.

An- und Auszeiten analysieren

Für jeden Marker wird bei jedem Bild abgespeichert, ob er gesehen wurde oder nicht. Hiermit kann man analysieren wie lange die An- und Auszeiten waren.

Es wird der Median der An-und Auszeiten berechnet. Das arithmetische Mittel wird nicht

genutzt, weil Ausreißer nicht zu sehr ins Gewicht fallen sollen.

$$s^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 \quad (3.5)$$

$$\bar{x} = \begin{cases} x_{m+1}, & \text{für ungerades } n = 2m+1 \\ \frac{1}{2}x_m + x_{m+1}, & \text{für gerades } n = 2m \end{cases} \quad (3.6)$$

Um zu verhindern, dass falsch erkannte Lichter zufällig einen richtigen Median haben, wird auch noch die empirische Varianz berechnet. Diese darf nicht zu hoch sein, damit das Licht auch als zuverlässig eingestuft wird.

Für beide Berechnungen wird jeweils die erste Anzeit und die aktuell laufende An- oder Auszeit nicht berücksichtigt. Bei der ersten Anzeit weiß man nicht wie lange das Licht schon vor der Detektion an war und ist dadurch nicht zuverlässig. Bei der letzten weiß man nicht wie lange es noch an bzw. aus sein wird.

Die genannten Werte werden außerdem nur dann berechnet, wenn der Marker mindestens zwei mal an und aus gewesen ist. Da man mindestens eine volle Anzeit und eine volle Auszeit zur Identifizierung benötigt.

Zugehörigen echten Marker finden

Um die erkannten Markern den Echten zuzuordnen, wurde eine Formel aus [3] genutzt, wobei i der Kandidat und j der echte Marker ist:

$$Wertung(i, j) = 1 - \frac{\text{expected time on}(j) - \text{mean time on}(i)}{\text{expected time on}(j)} \quad (3.7)$$

Insgesamt müssen folgende Punkte für die Identifizierung erfüllt sein:

1. Die Wertung muss mindestens 80 Prozent betragen
2. Der Median aller Auszeiten darf sich maximal um die Zeit eines Bildes von dem wahren unterscheiden
3. Der Median aller Anzeiten darf sich maximal um die Zeit eines Bildes von dem wahren unterscheiden
4. Die Varianz darf nicht zu hoch sein
5. Der Marker darf nicht länger als die Zeit der Auszeit nicht gesehen worden sein

3.3 Posenschätzung

Die Posenschätzung ist möglich, da man die reale Pose der Lichter kennt und weiß welche Lichter wo im Bild sind. Die Abbildung der Lichter im Bild gibt Auskunft in welcher

Richtung sich die Lichter befinden. Es lässt sich daraus aber nicht schließen wie weit die Lichter entfernt sind. Da man mehrere Lichter hat, von denen man die Richtung erkennen kann, die Tiefe aber nicht, wird die Lösung eindeutig.

Um die Pose zu schätzen, wurde ein Least-Squares Verfahren in dieser Art genutzt:

$$\min_x \frac{1}{2} \sum_i p_i(||f_i(x_{i_1}, \dots, x_{i_k})||^2) \quad (3.8)$$

f_i ist hierbei die Funktion, die die Kosten berechnet und p_i die Loss-Funktion. Zum Berechnen der Kosten wird ein von Sascha Arnold implementiertes Optical Feature Constraint genutzt. Hier wird die Kostenfunktion des minimum Least-Squares Verfahren definiert. Hierbei kann für jeden Marker der Pixel in dem er gesehen wurde, die reale Position, die Kovarianz und die intrinsischen Kalibrierungsparameter der Kamera übergeben werden um die Funktion zu berechnen. Als Loss-Funktion wird das SoftLOneLoss von Ceres¹ genutzt:

$$p(s) = 2(\sqrt{1+s} - 1) \quad (3.9)$$

Bei der Loss-Funktion ist sehr wichtig, dass Ausreißer nicht stark gewichtet werden. Durch Falscherkennungen kann es immer wieder zu solchen Ausreißern kommen, die natürlich nicht dadurch bedingt sind das sich ein einzelner Marker tatsächlich so weit von seiner projizierten Bildposition befindet. Die Position der falsch erkannten Marker steht dabei auch in keinem stochastischen Zusammenhang mit seiner eigentlichen Position. Das SoftLOneLoss gewichtet Ausreißer nicht besonders stark, deswegen wurde sich dafür entschieden. Das Optimierungsproblem wird mit einem Levenberg-Marquardt Algorithmus gelöst, der in Ceres verfügbar ist. Bei iterativen Verfahren braucht man immer eine bereits geschätzte Pose als Startwert. Hierfür wird in jedem Durchgang die zuletzt geschätzte gewählt. Für die erste Schätzung benötigt man aber eine "extern" geschätzte. Hierfür könnte zum Beispiel eine Posenschätzung aufgrund von akustischen Signalen dienen, wenn das AUV auch dadurch die Pose schätzt. Wenn eine geschätzte Pose mehr als 20 Meter entfernt ist, wird sie nicht als nächste initiale Schätzung gewählt und auch nicht veröffentlicht. Die Lichter sind nicht so weit zu sehen, daher weiß man das eine Entfernung von mehr als 20 Metern sehr unwahrscheinlich ist und es sich um einen Ausreißer handeln muss.

¹<http://ceres-solver.org/>

4 Hardware

4.1 Marker-Konfigurationen

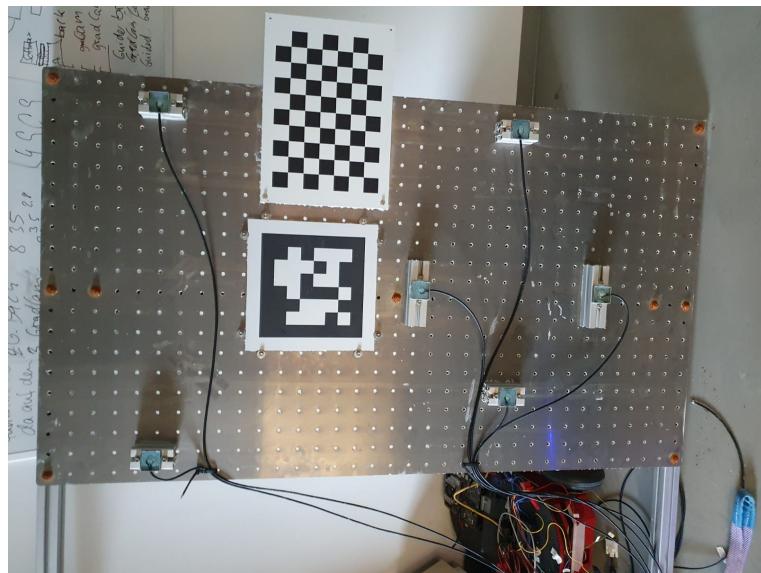


Abbildung 4.1: Konfiguration der Lichter

Welches Merkmal zum Identifizieren

Für die Marker-Konfiguration ist es essenziell einen Bestandteil zu haben, mit dem die Lichter voneinander unterschieden werden können. Für das Identifizieren, also letztendlich zum Unterscheiden der Marker könnten verschiedene Eigenschaften genutzt werden. Es können zum Beispiel unterschiedliche Blinkfrequenzen, unterschiedliche Farben oder eine spezielle Anordnung genutzt werden. Der Vorteil von Farben ist, dass ein Bild ausreicht, um alle Lichter zu erkennen und daraus eine Pose zu schätzen. Der Nachteil ist, dass Farben mit zunehmender Distanz unterschiedlich stark an Erkennbarkeit verlieren. Rot wird zum Beispiel deutlich mehr vom Wasser absorbiert als grün und blau. [17] In dieser Arbeit fiel die Entscheidung auf Blinkfrequenzen. Die Erkennung ist damit leider nur durch die Auswertung mehrerer Bilder möglich, durch die hohe Bildrate fällt diese

Verzögerung aber nicht besonders stark ins Gewicht. Ein großer Vorteil der Blinkfrequenzen ist, dass hiermit Fehler des Detektors korrigiert werden können. Wenn ein Licht als ein Marker identifiziert wird, muss es nicht nur wie ein Licht aussehen, sondern auch eine gewisse Blinkfrequenz aufweisen. So können falsch erkannte Lichter des Detektors, die meist nicht die richtige Blinkfrequenz aufweisen aussortiert werden.

Auswahl der Blinkfrequenzen

Die Wahl der Blinkfrequenzen sollte unter Berücksichtigung der Bildrate getätigt werden. Jede Anzeit und jede Auszeit müssen mindestens in einem Bild zu sehen sein. Um das zu gewährleisten, müssen die An- und Auszeit mindestens das Doppelte von der Zeit für ein Bild betragen. Dabei sollte gelten:

$$T_{m_0} = n_0 \cdot T_k \quad (4.1)$$

$$T_{m_1} = n_1 \cdot T_k \quad (4.2)$$

$$n_0, n_1 \in \mathbb{N} \quad (4.3)$$

T_k ist hier die Periode der Bildrate der Kamera, also der Zeit zwischen zwei Bildern. T_{m_0} und T_{m_1} repräsentieren die Aus- beziehungsweise Anzeit eines Markers. Sie sollten ganzzahligen Vielfachen der Periode der Bildrate entsprechen. Unter dieser Bedingung würden unabhängig von der vergangenen Zeit immer gleich viele Bilder gemessen werden, in denen der Marker aus oder an ist. Diese Bedingung macht die Messungen also gewissermaßen zeitinvariant. Es wurde in einer Rate von 20 Bilder pro Sekunde aufgenommen. Es wurden drei Blinkfrequenzen ausgesucht. Die erste Blinkfrequenz ist die kurz möglichste (Abbildung 4.2 (a)). Die zweite Blinkfrequenz (Abbildung 4.2 (c)) ist so gewählt, dass die Auszeit der Lichter möglichst gering gehalten wird. So ist die Verfolgung der Lichter robuster und zuverlässiger. So kann man auch in nahezu jedem Bild vier oder mehr Lichter erkennen. Die dritte Blinkfrequenz (Abbildung 4.2 (b)) verfolgt den gleichen Gedanken wie die zweite, aber mit leicht erhöhter Auszeit, um diese vermeintlich zuverlässiger zu erkennen.

Andere Merkmale

Helligkeit

Um so weit wie möglich erkannt zu werden, ist es natürlich ein Vorteil, wenn die Lichter möglichst hell sind. Für die Versuche waren die Lichter allerdings sowieso nicht weiter als drei Meter entfernt, weil das Becken keine größere Entfernung möglich machte. Aus geringer Entfernung kann eine zu große Helligkeit zu einem Problem werden, weil die Lichter sich dann teilweise überschneiden und auf dem Bild sehr groß werden. Dafür könnte man entweder die Belichtungszeit der Kamera verringern oder die Helligkeit niedriger stellen.

Für trübes Gewässer wurde eine höhere Helligkeit genutzt als für das klare Wasser.

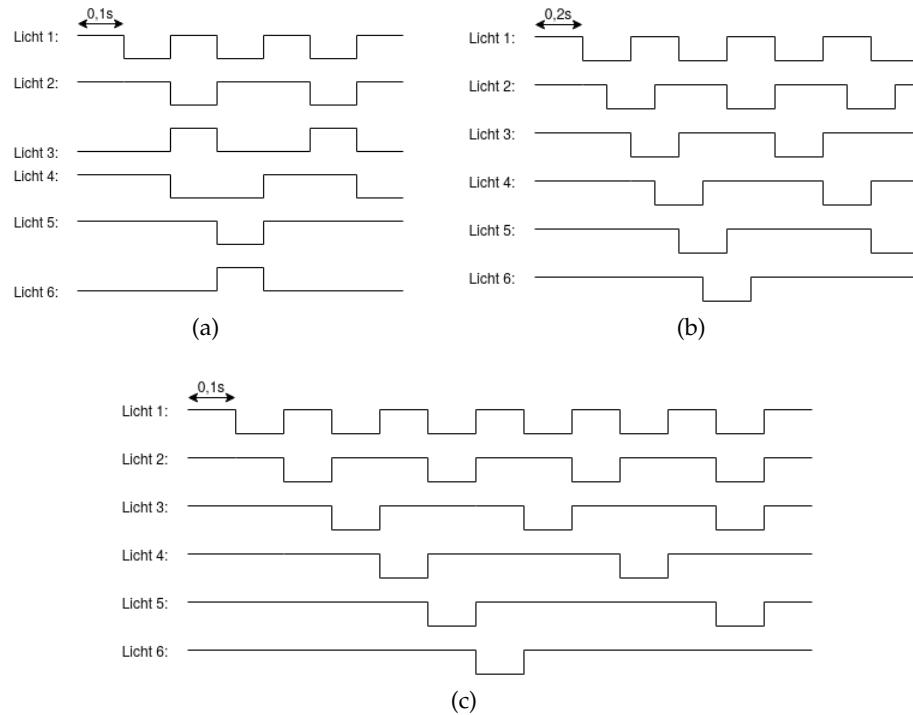


Abbildung 4.2: Blinkfrequenzen

Anordnungen

Bei der Anordnung sollen möglichst wenig kollineare Punkte entstehen. Dadurch verspricht man sich, dass bei egal welchen drei Punkten, man immer nur mit 3 identifizierten Punkten die Pose schätzen kann. Wenn drei Punkte zueinander fast kollinear sind, ist keine zuverlässige Posenschätzung garantiert. Damit wir also auch verschiedene Konfigurationen mit drei Lichten testen können wurde sich entschieden, die Licher möglichst nicht auf einer Linie zu positionieren.[19]

Anzahl

Ab drei Markern kann man die Pose bestimmen. [10]. Je mehr Marker es gibt, desto mehr Informationen kommen dazu und so kann die Erkennung noch robuster werden. In diesem Ansatz wurden insgesamt 6 Licher benutzt. Es wurden Konfigurationen mit drei, vier, fünf und sechs Lichern untersucht.

Abstand zwischen Lichern

Wenn man davon ausgeht, dass die Messungenauigkeiten und Fehler bei Erkennung des Mittelpunkts gleichverteilt sind, egal wie weit die Licher voneinander entfernt sind, kann man sagen, dass je weiter die Licher auseinander liegen, die Fehler immer unbedeutender

werden. Dadurch kann man davon ausgehen, dass es gut ist wenn die Lichter recht weit voneinander entfernt sind. Der Nachteil von einem größeren Abstand ist, dass man mehr Platz und Befestigung benötigt. Außerdem hat man eine größere Wahrscheinlichkeit das nicht alle Lichter gleichzeitig im Bild sind.

4.2 Technische Details der Lichter

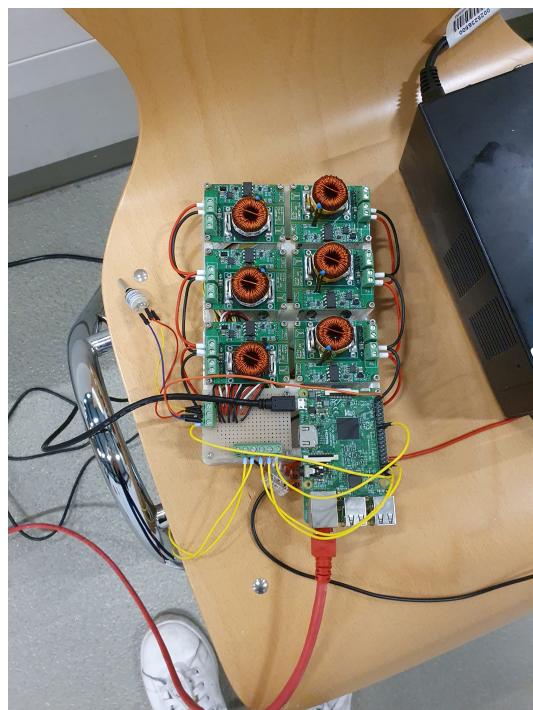


Abbildung 4.3: Hardware der Lichter

Die Lichter wurden durch einen Konstantstrom-Step-Down-Treiber (Abbildung 4.3) betrieben. Mit diesem konnten die Lichter an und ausgestellt werden und die Helligkeit der Lichter geändert werden. Es wurde an die Pins ein Raspberry Pi angeschlossen. Indem jeweils ein GPIO-Port des Raspberry Pis mit dem Pin des Treibers verbunden ist, der zum Auslösen des Lichts da ist. So konnten vom Raspberry Pi die GPIO-Ports angesteuert werden und somit die Lichter zum Blinken gebracht werden.

4.3 Technische Details der Kamera

Es wurde die Allied Vision Prosilica GC1380h digital Kamera genutzt. Diese hatte eine 1280x1024 Auflösung und es wurde eine 87 Grad Linse genutzt. Sie wurde mit dem prosilica_camera Treiber¹ genutzt.

¹http://wiki.ros.org/prosilica_camera

4.4 Versuchsaufbau

Versuch

Ort des Versuchs

Der Versuch wurde im Dunkelbecken des DFKIs gemacht. Das Becken ist $20\ m^3$ groß. Man kann es außerdem abdecken und es ist eintrübbbar. Die Maße sind $3,4\ m \times 2,6\ m \times 2,2\ m$. Bei dem Versuch wurde ein Gerät benutzt, an das die Kamera festgemacht werden konnte. Die Kamera konnte mithilfe des Geräts durch das Becken gefahren und um ihre Achse gedreht werden. So war es möglich mehrere Positionen und Rotationen zu testen.

Trübung des Wassers

Die Trübung wurde durch das Beifügen von Tonmehl erreicht. Mit Tonmehl können Eigenschaften von trübem Wasser nachgeahmt werden. [2] Leider stand kein Messgerät für die Trübung zur Verfügung. Deswegen wurde der Grad der Trübung nur durch die Menge des Tonmehls im Verhältnis zur Wassermenge berechnet. Die Trübung ist also mit einer gewissen Ungenauigkeit belastet.

Es wurden erst Aufnahmen im klaren Wasser gemacht.

Daraufhin wurde das Wasser immer mehr getrübt. Es wurden unter folgenden Trübwertwerten Aufnahmen gemacht: 2 FTU, 6FTU, 10 FTU, 12 FTU, 14 FTU und 16 FTU.

In den folgenden Bildern kann man erkennen, wie sich die Wasserqualität verändert hat.

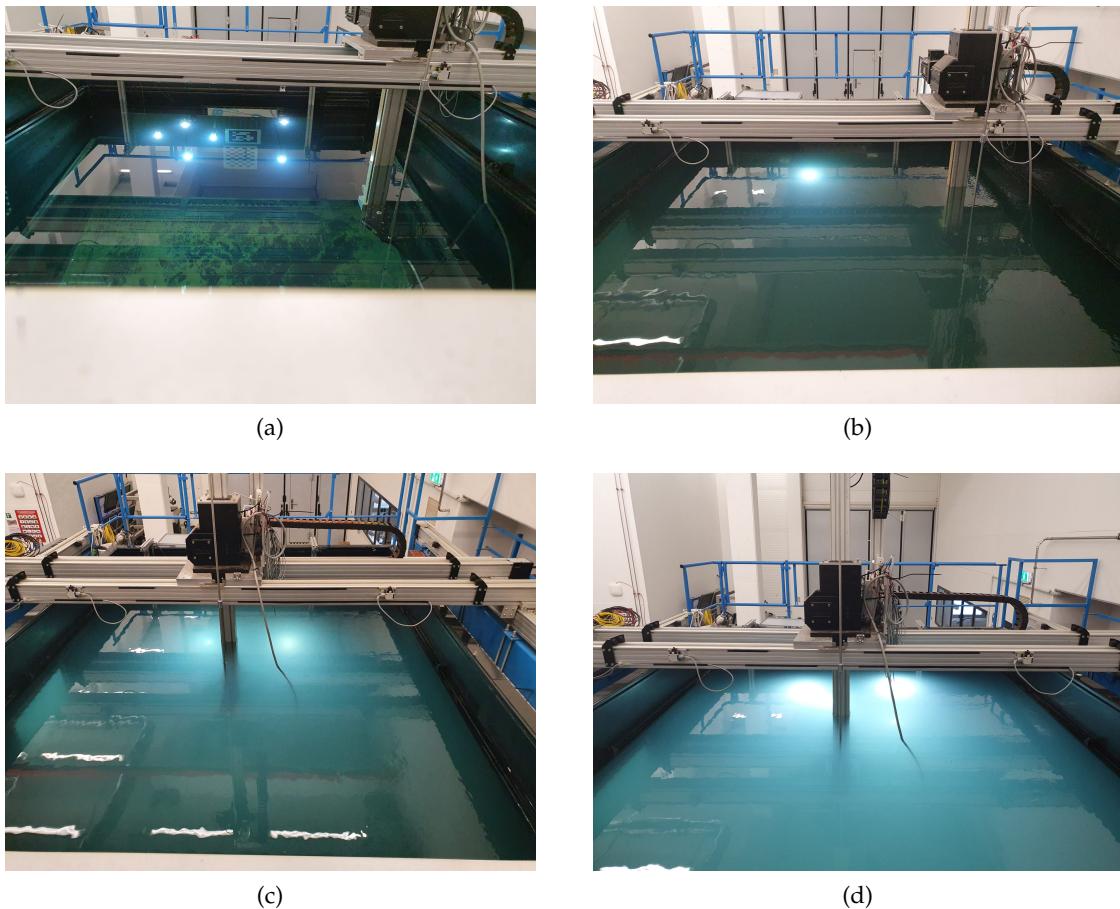


Abbildung 4.4: Wasser bei unterschiedlichen Trübungen

Posen

Im klaren Wasser wurden 9 verschiedene Posen untersucht. Drei der Posen waren so weit vorne, dass man gerade noch die Lichter im Bild hatte und die anderen sechs Posen waren so weit weg wie es ging. Bei drei der hinteren Posen war die Kamera außerdem tiefer im Wasser:

1. vorne links
2. vorne mitte
3. vorne rechts
4. hinten links
5. hinten mitte
6. hinten rechts
7. hinten links unten

8. hinten mitte unten

9. hinten rechts unten

In trübem Gewässer jeweils nur 4, da sich bei der Evaluation des trübem Wasser mehr auf den Detektor und weniger auf die Posenschätzung als ganzes fokussiert werden soll. Folgende Posen wurden untersucht:

1. vorne mitte

2. zentral mitte

3. hinten mitte

4. hinten mitte unten

5 Evaluation

5.1 Detektor

Um den Detektor zu evaluieren, wurde einmal der Recall und die Precision für verschiedene Posen der Kamera bestimmt. Die Precision gibt an, wie viel Prozent der erkannten Dinge richtig erkannt wurde. Hier steht TP für True Positive, gemeint sind hier also die Fälle, in denen etwas erkannt wurde, was auch wirklich da ist. FP steht für False Positive, gemeint sind hier die Fälle, wo etwas erkannt wurde, was allerdings gar nicht da ist.
Precision:

$$precision = \frac{|TP|}{|TP| \cup |FP|} \quad (5.1)$$

Der Recall gibt an, wie viel Prozent aller Dinge die erkannt werden sollen, auch erkannt werden. Ein hoher Recall heißt also, dass die allermeisten Marker erkannt wurden. FN steht für False Negative, gemeint sind hierbei die Dinge die erkannt werden sollten, es aber nicht wurden.

Recall:

$$recall = \frac{|TP|}{|TP| \cup |FN|} \quad (5.2)$$

Erkennung in klarem Wasser

	Pose 2	Pose 3	Pose 6
Precision	0.7350	0.8	0.3140
Recall	0.8820	1	1

Im klaren Wasser konnten die Lichter meistens gut erkannt werden. Allerdings kamen viele fälschlicherweise als Licht erkannte Punkte vor, wenn die Kamera näher an den Lichtern war, weswegen die Precision recht gering war. Bei Pose 2 und Pose 3 war die Kamera weiter entfernt von den Lichtern und es gab nicht so viele Störfaktoren, deswegen war die Precision hier recht hoch. Der Recall ist bei Pose 3 und 6 bei 100 Prozent, weil die Lichter recht gut erkennbar waren und so immer als Licht erkannt wurden. Bei Pose 2

gab es die Schwierigkeit, dass eins der Lichter über einem Wasserfleck lag und dadurch gebrochen wurde. So war das Licht sehr länglich auf der Kamera zu sehen (das kann man gut in Abbildung 5.1 (a) erkennen) und konnte nur erkannt werden wenn die Helligkeit aufgrund des Blinkens gerade niedriger war und das Licht so runder wirkte. Das Problem gab es auch bei mehreren Posen, oft konnte das Licht aber trotzdem gut erkannt werden wie in Pose 6 (Abbildung 5.1 (b)).

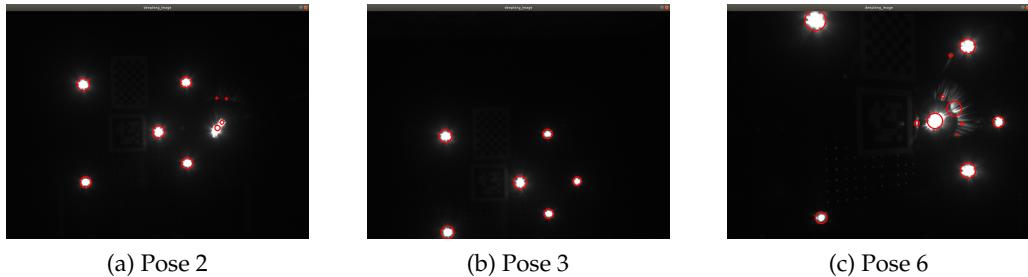


Abbildung 5.1: Erkennung des Detektors bei unterschiedlichen Posen

Erkennung bei Trübung

Der Detektor soll auch in trübem Wasser gut funktionieren. In der folgenden Tabelle kann man die Precision und Recall Werte von unterschiedlichen Trübungen bei einer Entfernung von ca. 2,5m zur Kamera sehen. Man kann erkennen, dass eine leichte Trübung sich sogar positiv auf das Ergebnis ausgeübt hat. Das lag daran, dass die Lichter (wie man in Abbildung 5.2 sieht) deutlich runder aussehen und andere Störfaktoren etwas ausgeblendet werden.

Bei 12 FTU konnte der Detektor die Lichter nicht mehr zuverlässig detektieren. Er hat die Lichter immer noch als Kandidaten gefunden und konnte die Mitte und Größe recht zuverlässig erkennen, die meisten Lichter wurden aber aufgrund der geringen Helligkeit als Lichter ausgeschlossen. Wenn man den Mindestwert für die Helligkeit des Lichtes heruntersetzen würde, könnte man bei 12 FTU, aber noch gut Lichter erkennen. Diese Werte wurden in dieser Arbeit nicht niedriger gewählt, weil sonst bei niedrigeren Entfernung und vor allem in klarem Wasser zu viele Dinge fälschlicherweise als Licht erkannt werden. Ab einem Trübungswert von 14 FTU wurden die Lichter aber auch nicht mehr als Kandidat erkannt.

	Trübung: 0FTU	Trübung: 6 FTU	Trübung: 10 FTU	Trübung: 12 FTU
Precision	0.80	0.9685	1	1
Recall	1	1	0.9343	0.3701

Entscheidung ob es sich um ein Licht handelt

Durch das teilweise stark abweichende Aussehen der Lichter, konnte die Filterung mit Hilfe der Radial Contrast Function nicht gut zum Aussortieren der Lichter dienen. Wenn

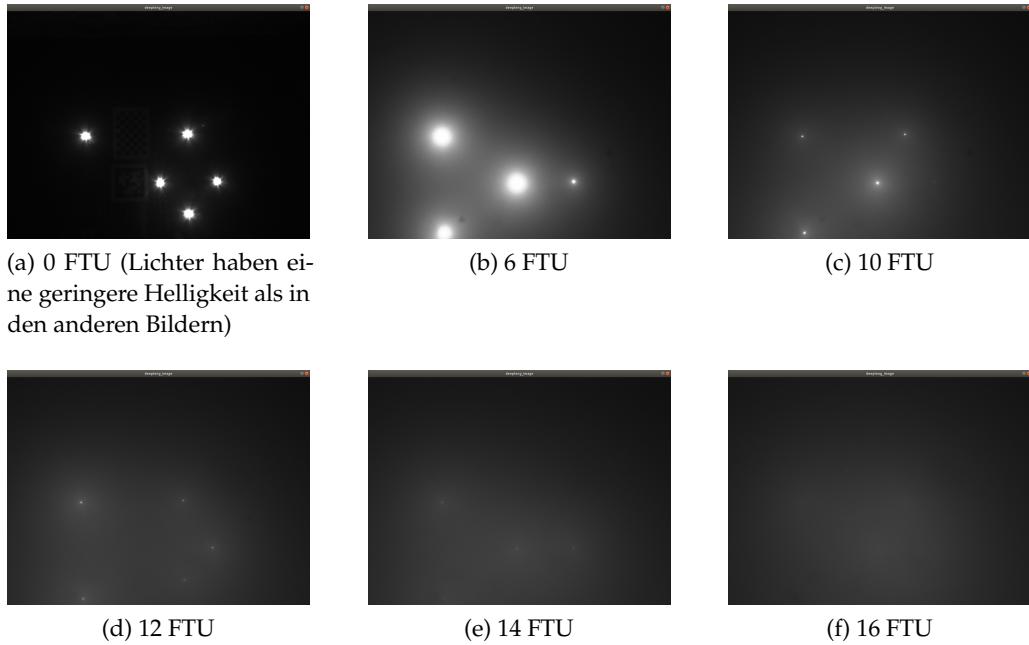


Abbildung 5.2: Kamerabild bei unterschiedlichen Trübungswerten

die Lichter recht klein im Bild zu sehen waren, war die Filterung mithilfe der Radial Contrast Function nützlich, um falsche Kandidaten auszusortieren, doch bei Lichtern, die größer erschienen und keine symmetrische Form hatten, wurden oft auch richtige Lichter herausgefiltert. Dies konnte man zwar durch Anpassen der Grenzwerte verhindern, dann wurden aber auch kaum noch falsche Lichter identifiziert.

Da durch den Identifizierer die meisten falschen Lichter als falsch deklariert wurden, wurde die Radial Contrast Function in der Auswertung nicht genutzt. Man könnte sie aber gut benutzen, wenn man weiß, dass man nicht so nah an die Lichter herangeht und die Lichter somit nie groß auf dem Kamerabild erscheinen. In den Versuchen gab es auch nicht so viele Störfaktoren, wenn das anders gewesen wäre, wäre die Funktion wahrscheinlich nützlich gewesen. Es könnte auch eine Lösung sein eine Variable einzuführen, mit der die Unsicherheit des Lichts bewertet wird. So wird je nachdem wie gut die Werte sind, das Licht als glaubwürdiger, oder unglaublich eingestuft.

Vor allem durch die minimale Helligkeit konnten die falschen Kandidaten gut gefiltert werden. Hier kam es aber auch vor, dass richtige Kandidaten als falsch identifiziert wurden, wenn das Wasser sehr trüb war und die Lichter kaum noch zu sehen. Der Schwellwert für die Helligkeit wurde aber trotzdem nicht angepasst, weil sonst an anderer Stelle zu viele Falsch-Positive entstanden wären.

5.2 Identifizierer

Beim Identifizieren wurden 3 Blinkfrequenzen getestet. Mit Allen war eine ähnlich gute Identifizierung möglich. Für die Auswertung des Identifizierers wird die schnellste Blinkfrequenz angesehen. Die Erstdetection dauert ab Erstdetection maximal:

$$2 \cdot t_{on} + t_{off} \quad (5.3)$$

Der Grund, wieso es zwei An-Perioden benötigt, bis das Licht identifiziert werden kann ist, dass man bei der ersten An-Periode nicht weiß wie lange das Licht schon vor Erstdetection an war und es damit nicht sicher ist, ob die man eine falsche Anzeit detektieren würde. Somit muss man auf die erste Anperiode nach einer Ausperiode warten, um einen zuverlässigen Wert zu haben. Da bei der Erstdetection das Licht immer aus und nie an ist, kann man bei der 1. Ausperiode sicher sein, dass sie komplett detektiert wurde.

Der Identifizierer baut auf den Detektor auf, das heißt, dass die Ergebnisse des Identifizierers abhängig vom Detektor sind. Der Identifizierer soll aber auch mit Fehldetections zurechtkommen. Um ihn zu evaluieren, werden einmal Pose sieben und Pose drei evaluiert. Bei Pose sieben kam es zu vielen Falsch-Positiven, die der Identifizierer ausschließen muss. Bei Pose 3 kam es zu wenigen Falsch-Positiven.

In der Tabelle sieht man, dass die Lichter bei beiden meistens identifiziert werden konnten, es gab aber auch immer wieder Fehlidentifikationen. Bei Pose 3 war das Problem, dass etwas von einem blinkenden Licht angeleuchtet wurde und auch als Licht detektiert wurde. So hatte es dieselbe Blinkfrequenz, wie das zu identifizierende Licht. Normalerweise wird bei so einem Fall das größere Licht bevorzugt, dies hat in 50 ausgewerteten Bildern aber bei 4 nicht funktioniert.

Bei Pose 7 gab es recht viele Fehldetections, die zu manchen Fehlidentifikationen geführt haben. Ein Problem des Identifizierers ist, dass er gerade bei der Erstdetection und kurz danach recht unzuverlässig ist, da es noch nicht so viele Daten zu den Blinkfrequenzen gibt. So kommt es gerade am Anfang oft zu vielen Ausreißern. Mit diesem Problem könnte man so umgehen, indem der Identifizierer das Licht mehrmals identifizieren muss, bevor man es als tatsächlich identifiziert annimmt.

	Pose 7	Pose 3
richtig erkannte	251	276
nach Erstdetection wieder verlorene	7	4
falsch identifizierte	9	4

5.3 Posenschätzer

Um den Posenschätzer zu evaluieren, wurde eine Ground Truth Pose benötigt. Die Ground Truth Pose wurde bestimmt, indem die Belichtungszeit sehr lang gewählt wurde und im hellen und klaren Wasser alle Posen angefahren wurden und mithilfe eines Apriltag-Markers und der Apriltag Erkennung von ROS die Posen bestimmt, werden konnten.

Später im trüben Wasser wurden die gleichen Posen angefahren, so gab es einen Vergleichswert.

Bei der Posenschätzung kann leider nicht nur der Posenschätzer selbst evaluiert werden, weil seine Ergebnisse auf den des Detektors und Identifizierers basieren. Wenn also der Mittelpunkt eines Lichts vom Detektor schlecht geschätzt wurde, hat das auch einen negativen Einfluss auf die Pose. Für die Evaluation wurden allerdings Posen gewählt in denen die Identifikation recht zuverlässig möglich war, um den Einfluss der Fehler des Detektors und Identifizierers so klein wie möglich zu halten. Es wurde einmal Pose 4 gewählt (Abbildung 5.6 (d)), die weiter von den Markern entfernt lag und Pose 7 (Abbildung 5.6 (g)), die eine geringere Entfernung hat. Bei Pose 4 sieht man, dass die Posenschätzung am Anfang recht schlecht funktioniert. Das liegt daran, dass der Identifizierer bei Erstidentifizierung recht ungenau ist und es häufig zu Fehlidentifikationen kommt. Die Fehler kommen daher vom Identifizierer und nicht vom Posenschätzer. Sonst kann man in den Diagrammen erkennen, dass die Posenschätzung mit 6 Lichtern gut funktioniert. Die größten Fehler kommen an der z-Achse zustande. Die z-Achse ist hier die Achse, die von der Kamera aus nach vorne geht, also meistens den höchsten Wert hat. Gerade hier können schnell Fehler bei Messungenauigkeiten auftreten. Bei nur 3 Lichtern und einer Konfiguration mit 4 Lichtern hat die Posenschätzung allerdings nicht so gut geklappt. Es sollten also mindestens 4 Lichter zu sehen sein.

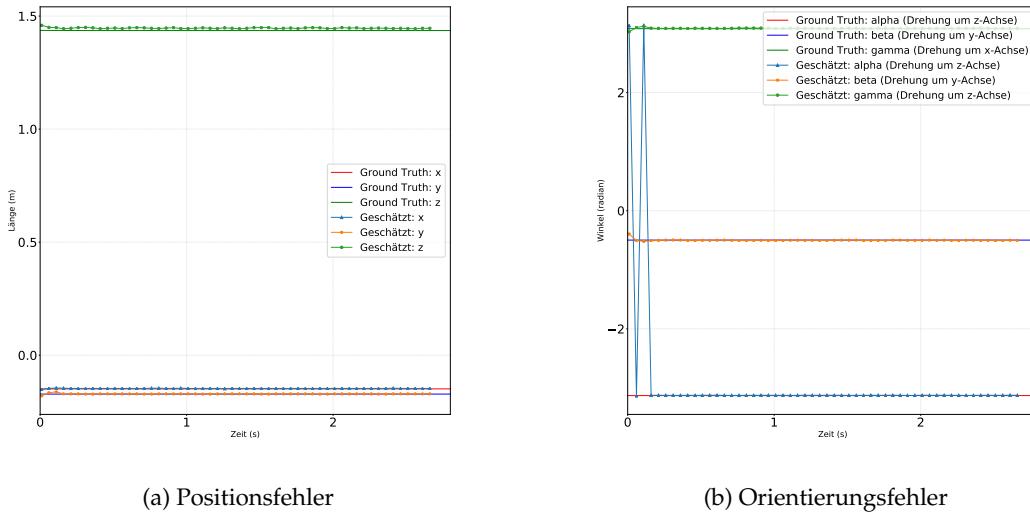


Abbildung 5.3: Pose 7

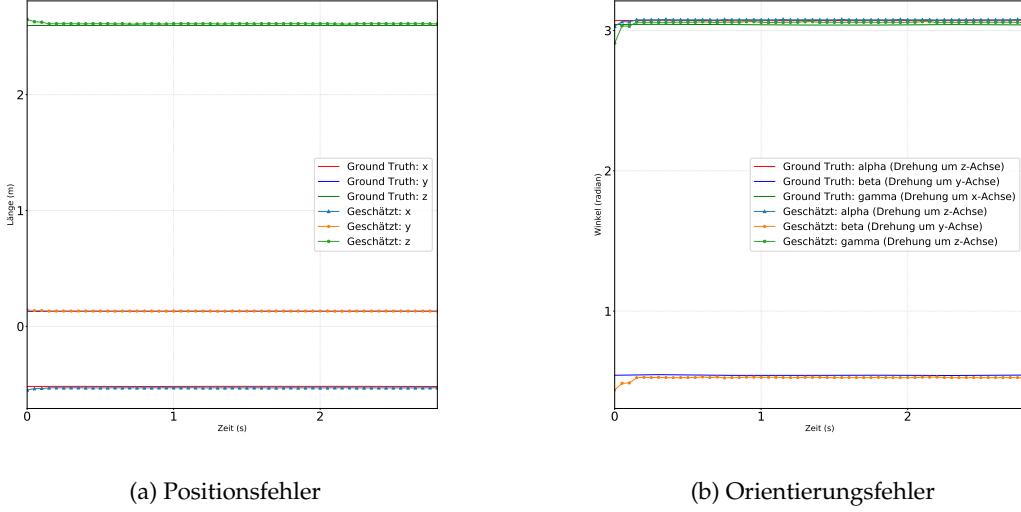


Abbildung 5.4: Pose 4

5.4 Unterschiede zwischen Markerkonfigurationen

Im Folgenden werden die Markerkonfiguration evaluiert. Hierbei sollen verschiedene Ansätze zur Konfiguration der Marker miteinander verglichen werden. Durch die Auswertung verschiedener Ansätze kann bestimmt werden, welche sich bewähren. Hierfür wurden bei der Schätzung der Pose nicht alle Lichter berücksichtigt, sondern immer unterschiedliche Gruppen. So kann die Posenschätzung auf verschiedene Anzahl, Abstand und Anordnung der Lichter getestet werden. Im Folgenden werden die unterschiedlichen Lichter in Zahlen von 1-6 unterteilt. (siehe Abbildung 5.5).

Es folgen Grafiken über den Mittelwert und der Standardabweichung des Positionsfehlers und Orientierungsfehlers an den unterschiedlichen Positionen der Kamera. Um den Fehler zwischen den Positionen darzustellen, wird als Metrik der euklidische Abstand d zwischen der geschätzten Position pos_{est} und der Ground Truth Position pos_{gt} genutzt:

$$d(pos_{gt} - pos_{est}) = \|pos_{gt} - pos_{est}\|_2 \quad (5.4)$$

Um den Fehler zwischen den Orientierungen darzustellen, wird als Metrik die Angle-Axis-Repräsentation verwendet: [12]

$$\phi(R_{gt}, R_{est}) = \|\log(R_{gt}R_{est}^T)\| \quad (5.5)$$

In den folgenden Grafiken sieht man den Mittelwert und die Standardabweichung des Fehlers an den verschiedenen Posen der Kamera, für die unterschiedliche Konfigurationen. Für jede mögliche Anzahl an Markern gibt es eine eigene Grafik. Wobei dies bei einer Anzahl von 3 beginnt, weil bei einer kleineren Anzahl noch keine zuverlässige Schätzung möglich ist.

Auffällig ist, dass die Standardvarianz je nach Pose sehr variiert und manchmal sehr

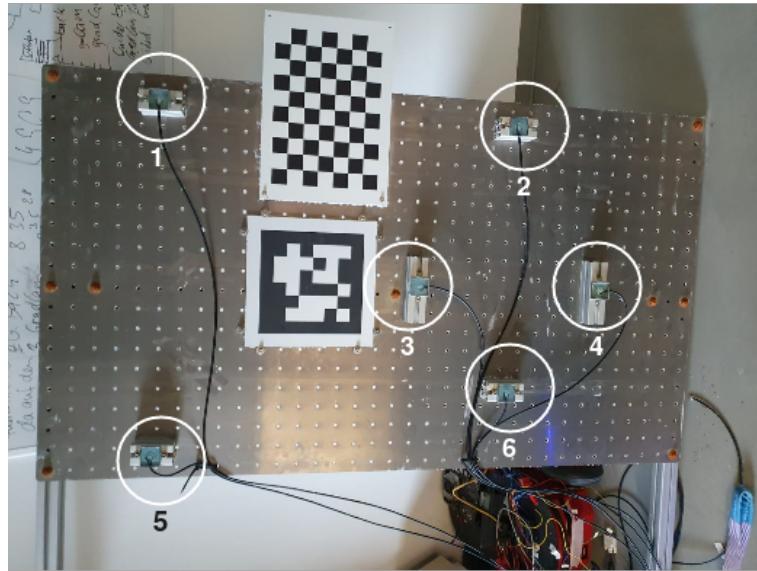


Abbildung 5.5: Numerierung der Lichter

gering und manchmal sehr hoch ist. Das liegt daran, dass bei manchen Konfigurationen und Posen der Kamera die Lichter fast nie falsch identifiziert wurden, oder wenn dann nur eins zur selben Zeit. Bei Pose 9 zum Beispiel kam es häufig zu Fehlidentifikationen, deswegen gibt es bei fast jeder Konfiguration eine starke Standardabweichung. Man kann klar erkennen, dass der Fehler bei der Schätzung besser wird, je mehr Marker einbezogen werden. Hierbei lässt sich aber auch erkennen, dass bei steigender Lichtanzahl der Unterschied immer weniger signifikant wird. Der Unterschied zwischen 3 und 4 Lichtern ist recht eindeutig, zwischen 5 und 6 Lichtern kann man aber kaum noch eine Verbesserung erkennen. Ein weiterer Vorteil von mehr als 3 Lichtern ist, dass die Pose, auch wenn ein Licht gerade nicht im Bild ist oder verdeckt immer noch geschätzt werden kann. Bei einer Anzahl von 3 Markern hat in den untersuchten Posen das Dreieck '135' am besten abgeschnitten. Ein Grund hierfür könnte sein, dass die Marker im Vergleich zu den Dreiecken '234' und '246' recht weit auseinander liegen.

Bei den Konfigurationen mit 4 Markern gab es bei der Konfiguration '2346' deutlich größere Fehler in der Schätzung als bei beiden anderen Konfigurationen. Hier könnte auch der Abstand der Lichter zueinander eine Rolle spielen. Man kann beobachten, dass die Konfigurationen '1256' und '1245' beide bessere Werte als Konfiguration '2346' haben und beide mehr Abstand zwischen den Markern haben. Bei den Konfigurationen mit insgesamt 5 Lichtern, gab es keine starken Unterschiede mehr zwischen den Konfigurationen, am besten schnitt aber Konfiguration '12356' ab. Bei der Anzahl von 6 Lichtern gab es nur eine Konfiguration, da nur 6 Lichter genutzt wurden. Hier kann man keine Verbesserung zu den Konfigurationen mit 5 Lichtern beobachten. Die Konfiguration '12356' hat sogar bessere Werte als die Konfiguration mit 6 Lichtern.

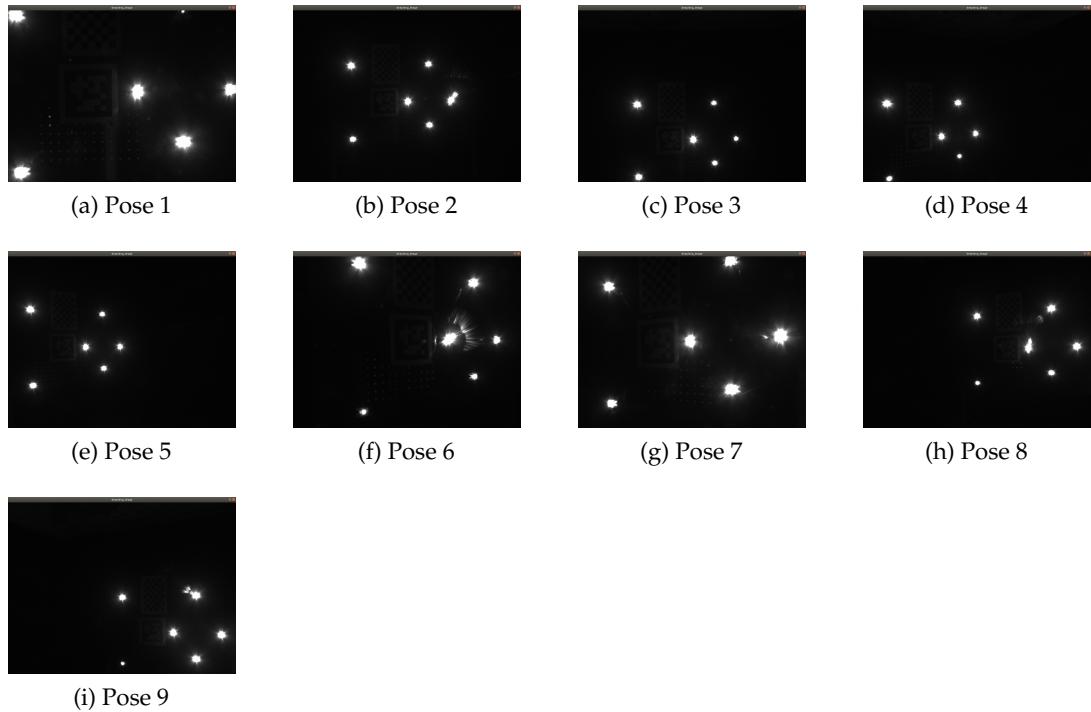


Abbildung 5.6: Posen der Kamera

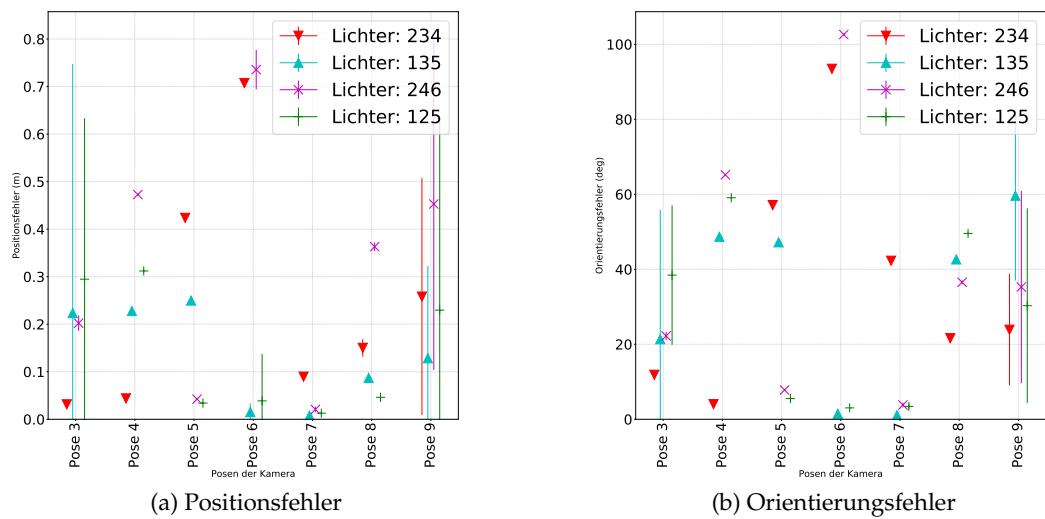


Abbildung 5.7: Anzahl Lichter: 3

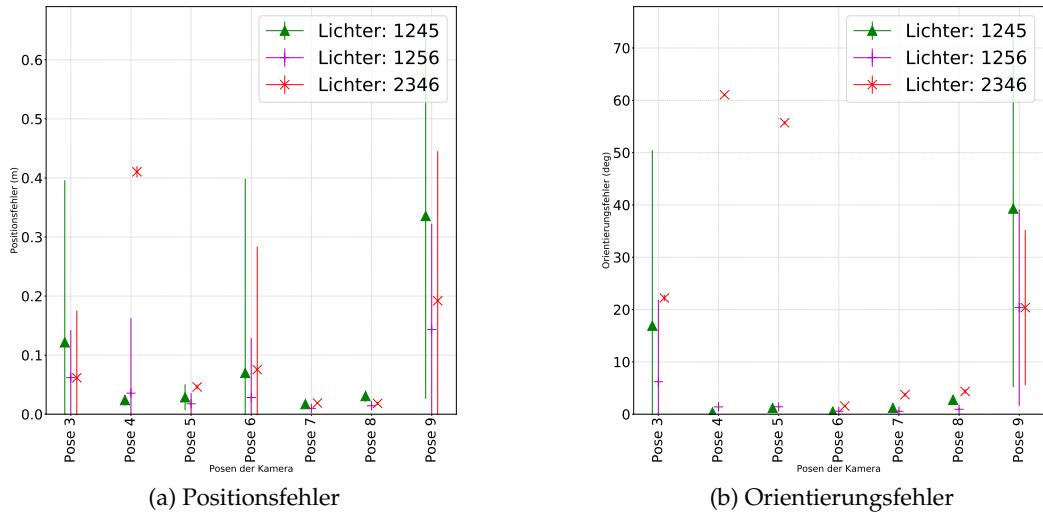


Abbildung 5.8: Anzahl Lichter: 4

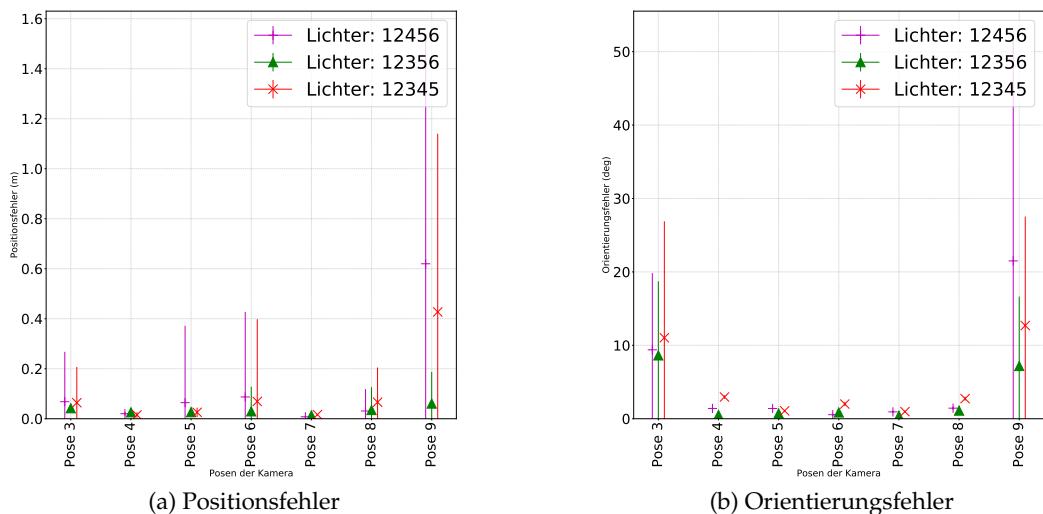


Abbildung 5.9: Anzahl Lichter: 5

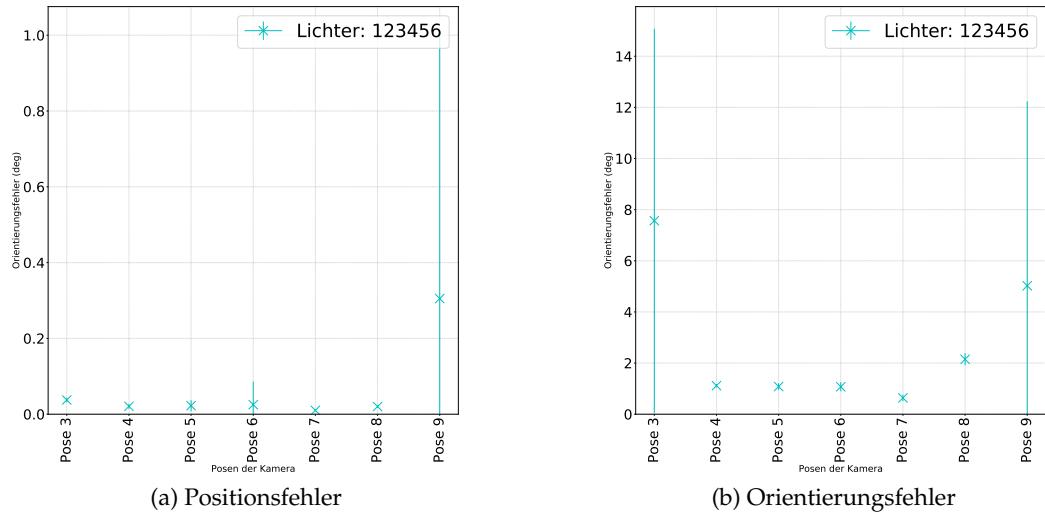


Abbildung 5.10: Anzahl Lichter: 6

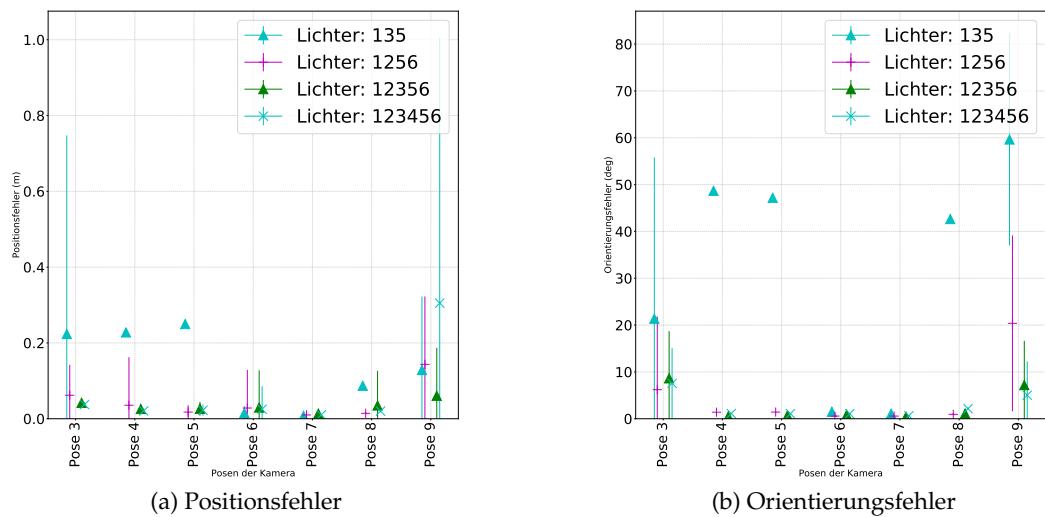


Abbildung 5.11: Beste Konfiguration ihrer Anzahl im Vergleich

Blinkfrequenzen

Alle 3 Blinkfrequenzen konnten recht zuverlässig erkannt werden. Der Vorteil von den beiden längeren Blinkfrequenzen ist, dass die Auszeit im Verhältnis recht kurz ist, das heißt, dass man das Licht häufiger sieht und so Veränderungen besser sehen kann und mehr Daten für die Schätzung der Pose zur Verfügung stehen.

Bei der schnellsten Frequenz ist der Vorteil, dass die Lichter sehr schnell identifiziert werden können.

6 Fazit

Im Rahmen der Arbeit wurden unterschiedliche Konfigurationen von aktiven Markern erstellt, um daraus die Pose schätzen zu können. Es wurde ein Ansatz implementiert, mit dem die Marker im Kamerabild erkannt und identifiziert wurden, sodass die Pose bestimmt werden konnte. Zum Ende konnten verschiedene Konfigurationen miteinander verglichen werden und man konnte sehen welche Konfigurationen sich besser eignen und welche nicht.

In den Versuchen hat die Erkennung bei guten Bedingungen zuverlässig funktioniert. Erst bei starker Verschmutzung, bei denen die Marker nur sehr eingeschränkt erkennbar waren, gab es eine hohe Rate an nicht erkannten Markern. AUVs haben normalerweise Scheinwerfer, die nach vorne strahlen um auf der Kamera mehr zu erkennen. Es ist zu erwarten, dass die Kamerabilder heller und anfälliger für Fehldetektionen sind als die in den Versuchen aufgenommenen. Hier müsste der Detektor also angepasst werden, um nicht zu viele Falsch Positive zu erkennen. Der Ansatz der Radial Contrast Function könnte hier zur Filterung genutzt werden.

Für die Mehrzahl der untersuchten Posen konnten die Marker sehr zuverlässig identifiziert werden. Es lässt sich davon ausgehen das die Auswirkung des Identifizierers auf den Fehler vernachlässigbar sind. Einige Posen zeigen dabei Fehler in deutlich höheren Größenordnungen. Es ist davon auszugehen, dass der Identifizierer Fehlerkennungen des Detektors nicht korrigieren konnte. Marker sind hier nicht oder falsch identifiziert worden. Solche Ausreißer müssten in der Anwendung des Algorithmus kompensiert werden um die Pose hinreichend genau im Verlauf eines Andockmanövers oder Ähnlichem bestimmen zu können.

Der Posenschätzer hat bei richtiger Identifikation der Marker bei den meisten Konfigurationen sehr zuverlässig funktioniert. Bei einer Konfiguration mit vier unterschiedlichen Markern und allen Konfigurationen mit nur drei Markern, konnte die Pose aber nicht hinreichend genau geschätzt werden, auch wenn alle richtig identifiziert und detektiert wurden. Bei den anderen Konfigurationen konnten einzelne Fehlidentifikationen gut ausgeglichen werden.

Es konnten verschiedene Konfigurationen miteinander verglichen werden. Man konnte sehen wie sich die Posenschätzung bei höherer Markeranzahl verbesserte, wobei dies nur bis zu einer Markeranzahl von fünf auftrat. Zwischen fünf und sechs Markern konnte man keinen Unterschied mehr erkennen. Man konnte auch sehen, dass bei Konfigurationen mit selber Anzahl von Markern mit unterschiedlichen Anordnungen die Posenschätzung unterschiedlich präzise war. Im Rahmen der Arbeit konnte aber kein klarer Grund für die

Unterschiede identifiziert werden.

Anhang

Der Anhang befindet sich auf dem Datenträger inklusive einer digitalen Auflistung der vorhandenen Daten und Dokumente.

Alternativ kann der Quellcode auch unter folgendem Link eingesehen werden: https://github.com/jotholen/BA_tholen

Literaturverzeichnis

- [1] Sabine Achilles. Markerloses tracking unter verwendung von analyse durch synthese auf basis von featuredetektoren. 2008.
- [2] Jan Albiez, Alexander Duda, Martin Fritzsche, Felix Rehrmann, and Frank Kirchner. Csurvey—an autonomous optical inspection head for auvs. *Robotics and Autonomous Systems*, 67:72–79, 2015. Advances in Autonomous Underwater Robotics.
- [3] Josep Bosch, Nuno Gracias, Pere Ridao, Klemen Istenič, and David Romagós. Close-range tracking of underwater vehicles using light beacons. *Sensors (Basel, Switzerland)*, 16, 03 2016.
- [4] Frank Carsey, GunShing Chen, James Cuns, Lloyd French, Roger Kern, A. Lane, Paul Stolorz, Wayne Zimmerman, and Phil Ballou. Exploring europa’s ocean: A challenge for marine technology of this century. *Marine Technology Society Journal*, 33:5–12, 01 1999.
- [5] Elan Dubrofsky and Robert J Woodham. Combining line and point correspondences for homography estimation. In *International Symposium on Visual Computing*, pages 202–213. Springer, 2008.
- [6] Tomio Echigo. A camera calibration technique using three sets of parallel lines. *Machine Vision and Applications*, 3(3):159–167, 1990.
- [7] Sascha Flögel, Detlef Wilde, Ingo Ahrns, Christof Nuber, Marc Hildebrandt, Alexander Duda, and Jakob Schwendner. A new deep-sea crawler system - mansio-viator. In *Proceedings of the Oceans 18 MTS/IEEE Kobe Techno-Ocean2018. OCEANS MTS/IEEE Conference (OCEANS-2018), Ocean Planet - its our home, May 28-31, Kobe, Japan*. IEEE, IEEE, 2018.
- [8] Christian Gaida. Untersuchung von verfahren zur pose-schätzung im hinblick auf analyse durch synthes. 2011.
- [9] Nuno Gracias, Josep Bosch, and Mohammad Karim. Pose estimation for underwater vehicles using light beacons. *IFAC-PapersOnLine*, 48:70–75, 12 2015.
- [10] Bert M Haralick, Chung-Nan Lee, Karsten Ottenberg, and Michael Nölle. Review and analysis of solutions of the three point perspective pose estimation problem. *International journal of computer vision*, 13(3):331–356, 1994.

- [11] Marc Hildebrandt, Jan Albiez, Martin Fritsche, Jens Hilljegerdes, Philipp Kloss, Marius Wirtz, and Frank Kirchner. Design of an autonomous under-ice exploration system. In *2013 OCEANS-San Diego*, pages 1–6. IEEE, 2013.
- [12] Du Q. Huynh. Metrics for 3d rotations: Comparison and analysis. *Journal of Mathematical Imaging and Vision* 35, 35:155–164, 03 2009.
- [13] Vincent Lepetit and Pascal Fua. *Monocular model-based 3D tracking of rigid objects*. Now Publishers Inc, 2005.
- [14] Khinnwe Lwin, Naoki Mukada, Myo Myint, Daiki Yamada, Mamoru Minami, Takayuki Matsuno, Kazuhiro Saitou, and Waichiro Godou. Docking at pool and sea by using active marker in turbid and day/night environment. *Artificial Life and Robotics*, 23:1–11, 06 2018.
- [15] Marc Masias Moyset. Automatic source detection in astronomical images. 2014.
- [16] Edwin Olson. Apriltag: A robust and flexible visual fiducial system. In *roceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE,, page 3400–3407, 2011.
- [17] Yan-Tsung Peng and Pamela C. Cosman. Underwater image restoration based on image blurriness and light absorption. *IEEE Transactions on Image Processing*, 26(4):1579–1594, 2017.
- [18] Emanuele Trucco and Alessandro Verri. *Introductory techniques for 3-D computer vision*, volume 201. Prentice Hall Englewood Cliffs, 1998.
- [19] Yuko Uematsu and Hideo Saito. Ar registration by merging multiple planar markers at arbitrary positions and poses via projective space. In *Proceedings of the 2005 International Conference on Augmented Tele-Existence, ICAT ’05*, page 48–55, New York, NY, USA, 2005. Association for Computing Machinery.
- [20] Richard Hartley Andrew Zisserman. Multiple view geometry in computer vision. 2004.