

UNIVERSITÄT BIELEFELD
TECHNISCHE FAKULTÄT
SOMMERSEMESTER 2016

Underwater Image Quality Measure

Projekt „Underwater Image Understanding“ (392167)

Dozent: Prof. Dr.-Ing. Tim Wilhelm Nattkemper

Betreuer: Torben Möller

Autor	Thomas Krahn
Studiengang	Naturwissenschaftliche Informatik (Master)
Matrikelnummer	2087261
E-Mail-Adresse	tkrahn@techfak.uni-bielefeld.de

Inhaltsverzeichnis

1. Einleitung	1
2. Methodik	2
2.1. Grundlagen des Algorithmus' zur automatischen Qualitätsbewertung von Unterwasserfotos	2
2.2. Implementierung des Algorithmus' mittels C++ und OpenCV	6
2.3. Integration in Copria	6
3. Ergebnisse und Diskussion	6
3.1. Vergleich der Ergebnisse des selbst implementierten Algorithmus' mit denen des Vorgegebenen	6
3.2. Evaluation des Algorithmus' mithilfe manuell sortierter Unterwasserfotos	10
3.3. Verbesserungs- und Anpassungsmöglichkeiten	10
4. Fazit	11
Literatur	12
A. Anhang	13
A.1. Fotos der Tiefseestation HAUSGARTEN zur Überprüfung des Algorithmus'	13

1. Einleitung

Bei der Erforschung der Unterwasserwelt kommen viele bildgebende Verfahren zum Einsatz. Insbesondere bei der Untersuchung der Tiefsee werden auch Roboter mit Kamerasystemen genutzt. Die Unterwasserfotografie ist aufgrund der physikalischen Eigenschaften von Wasser jedoch eine große Herausforderung, denn Aborptions- und Streuungseffekte des Wassers schränken die Sichtbarkeit von Unterwasserobjekten stark ein [1]. Aus dem Grund weisen Unterwasserfotos häufig geringen Kontrast, eine ungleichmäßige Beleuchtung, Verschleierungen, Lichtartefakte, zerstörte Farben, Rauschen und andere Störungen auf.

Es wurden bisher einige Methoden entwickelt, die Unterwasserfotos durch aufwändige Algorithmen verbessern [1]. Meist ist jedoch eine manuelle Beurteilung notwendig, die zeit- und arbeitsaufwändig ist. Aus dem Grund sind verlässliche objektive Evaluationsmessungen von Vorteil. Im Wesentlichen lassen sich die objektiven Messmethoden in zwei Kategorien teilen. Abhängig davon, ob ein ideales Bild verfügbar ist, auf dessen Grundlage die Bewertung des Bildes stattfinden kann, kann entweder auf *full-reference* oder *non-reference measures* zurückgegriffen werden. Da für Unterwasserfotos im Allgemeinen keine idealen Vergleichsbilder vorhanden sind, wird in diesem Fall auf die *non-reference*-Methode zurückgegriffen werden.

Aus diesem Grund wird in dieser Ausarbeitung im Rahmen des Projektes „*Underwater Image Understanding*“, das im Sommersemester 2016 an der Universität Bielefeld stattgefunden hat, ein *non-reference measure* Algorithmus zur automatischen Qualitätsbewertung von Unterwasserfotos vorgestellt, der auch implementiert wurde. Er soll bei der automatischen Sortierung von Unterwasserfotos nach Qualität Verwendung finden, um den Forschern bei der Betrachtung von Unmengen an Bilddaten Fotos zu ersparen, die eine geringe Bildqualität aufweisen.

2. Methodik

2.1. Grundlagen des Algorithmus' zur automatischen Qualitätsbewertung von Unterwasserfotos

Der von Panetta et al. [1] veröffentlichte Algorithmus, die sogenannte *underwater image quality measure* (UIQM), bietet die Grundlage für dieses Projekt. Ziel des Algorithmus' ist, eine hohe Korrelation zwischen der menschlichen Wahrnehmung und der automatisch beurteilten Bildqualität zu erreichen [1]. Der Algorithmus beruht auf drei Metriken, die im Folgenden vorgestellt werden sollen.

Underwater image colorfulness measure (UICM) Mit zunehmender Wassertiefe wird das Licht abhängig von der Wellenlänge abgeschwächt [2]. Bereits bei einer Tiefe von ca. 3 m verschwindet die Farbe rot. Bei einer Tiefe von ca. 5 m geht orange verloren. Nachdem bei ca. 10 m gelb nach und nach verschwindet, dringen die noch kurzwelligeren Lichtstrahlen noch tiefer bis zuletzt auch blau verschwindet. Da rot aufgrund der größten Wellenlänge als erstes absorbiert wird, haben die meisten Unterwasserfotos einen blau-grünen Charakter.

Untersuchungen haben gezeigt, dass man zur Bestimmung der Farbigkeit gut mit statistischen Werten arbeiten kann. Panetta et al. [1] nutzen aufgrund starker Bildartefakte Ausreißerbereinigte Statistiken. Sie berechnen statische Mittelwerte für die gegensätzlichen Farbkomponenten RG und YB, die definiert sind durch

$$RG = R - G \quad (1)$$

$$YB = \frac{R + G}{2} - B, \quad (2)$$

wobei R, G und B die Rot-, Grün- und Blauwerte eines Pixels bezeichnen. Für ein Bild der Größe $M \times N$ sei $K = M \cdot N$ die Anzahl der Pixel sowie $T_{\alpha_L} = \lceil \alpha_L K \rceil$ (die nächste Zahl größer oder gleich $\alpha_L K$) und $T_{\alpha_R} = \lfloor \alpha_R K \rfloor$ (die nächste Zahl kleiner oder gleich $\alpha_R K$). Dann ist der Ausreißer-bereinigte Mittelwert jeweils definiert als

$$\mu_{\alpha, RG} = \frac{1}{K - T_{\alpha_L} - T_{\alpha_R}} \sum_{i=T_{\alpha_L}+1}^{K-T_{\alpha_R}} \text{Intensity}_{RG_i} \quad (3)$$

$$\mu_{\alpha, YB} = \frac{1}{K - T_{\alpha_L} - T_{\alpha_R}} \sum_{i=T_{\alpha_L}+1}^{K-T_{\alpha_R}} \text{Intensity}_{YB_i}, \quad (4)$$

wobei die Bildpixel jeweils sortiert sind bezüglich der RB- bzw. YB-Werte, so dass $RG_1 \leq RG_2 \leq \dots \leq RG_K$ und $YB_1 \leq YB_2 \leq \dots \leq YB_K$.

Der Ausreißer-bereinigte Mittelwert bietet die Möglichkeit der Anwendung auf verschiedenen Typen von Bildern [1]. Abbildung 1(a) zeigt den Nutzen von $\alpha_R > 0$ in den Fällen, in denen sehr helle Regionen wie Luftblasen oder Lichtstrahlen im Bild vorhanden sind, die aber eigentlich nicht zum zentralen Bildinhalt gehören. Standardmäßig wird $\alpha_R = \alpha_L = 0,1$ genutzt.

Ein Mittelwert nahe an null deutet auf einen besseren Weißabgleich hin, d.h. dass keine der



Abbildung 1: (a) Unterwasserfoto mit Luftblasen. Bei einem derartigen Bild sollte $\alpha_R > 0$ gewählt werden, um die Luftblasen für den Mittelwert nicht zu beachten. (b) Rauschfreies Unterwasserfoto, weshalb sowohl $\alpha_L = 0$ als auch $\alpha_R = 0$ gewählt werden sollten [1].

Tabelle 1: Statistische Werte für die Fotos in Abbildung 2.

	μ_{RG}	μ_{YB}	σ_{RG}	σ_{YB}	UICM
Abbildung 2(a)	-10,8300	61,8086	13,1411	11,0085	-0,5889
Abbildung 2(b)	0,7896	0,8901	23,9203	19,9269	4,9058

Farben dominant ist [1]. Wie Tabelle 1 zeigt, hat Abbildung 2(a) einen deutlich niedrigeren Mittelwert als Abbildung 2(b). Offensichtlich sind die Farben in 2(b) deutlich ausgeglichener.

Die Varianzen werden berechnet durch

$$\sigma_{\alpha,RG}^2 = \frac{1}{N} \sum_{p=1}^N \left(\text{Intensity}_{RG_p} - \mu_{\alpha,RG} \right)^2 \quad (5)$$

$$\sigma_{\alpha,YB}^2 = \frac{1}{N} \sum_{p=1}^N \left(\text{Intensity}_{YB_p} - \mu_{\alpha,YB} \right)^2 \quad (6)$$

und lassen Rückschlüsse auf die Dynamik im Bild zu. Je größer die Varianz, desto größer die Dynamik und somit die Unterschiede zwischen den Farben. Wie Tabelle 1 zeigt, sind auch die Varianzen sowohl für die RG- als auch die YB-Komponente für das farbkorrigierte Foto größer.

Das Underwater image colorfulness measure (UICM) setzt sich nun wie folgt aus den (Ausreißerbereinigten) Statisiken zusammen [1]:

$$\text{UICM} = -0,0268 \sqrt{\mu_{\alpha,RG}^2 + \mu_{\alpha,YB}^2} + 0,1586 \sqrt{\sigma_{\alpha,RG}^2 + \sigma_{\alpha,YB}^2} \quad (7)$$

Die Koeffizienten wurden über lineare Regression ermittelt. Je größer dieser Term für ein Foto wird, desto besser ist die Farbigkeit des Fotos.

Underwater image sharpness measure (UISM) Durch Streuungseffekte gehen bei der Aufnahme von Unterwasserobjekten viele Details verloren. Aus diesem Grund bestimmt die zweite Metrik die Schärfe eines Bildes. Um diese zu messen, wird der Sobel-Kanten-Detektor auf jede der RGB-Komponenten angewendet und das resultierende „Kantenbild“ mit dem Originalbild multipliziert. Dadurch bleiben nur die Kantenpixel des Originalbildes erhalten.

Mathematisch kann der UISM-Wert wie folgt berechnet werden, wobei das Bild in $k_1 k_2$ Blöcke aufgeteilt wird und $I_{\max,k,l}$ den maximalen und $I_{\min,k,l}$ den minimalen Intensitätswert innerhalb



UICM = -0,5889



UICM = 4,9058

Abbildung 2: (a) Originales Unterwasserfoto. (b) Unterwasserfoto nach Farbkorrektur. Offensichtlich hat das originale Foto einen starken grünen Charakter, während das farbkorrigierte Foto deutlich natürlichere Farben aufweist. Entsprechend ist der UICM-Wert (siehe Formel (7)) für das farbkorrigierte Foto höher [1].



UISM = 0.5940



UISM = 3.0707

Abbildung 3: (a) Originales Unterwasserfoto. (b) Unterwasserfoto nach Streuungskorrektur. Der UISM-Wert (siehe Formel (9)) ist nach Anwendung eines Korrektur-Algorithmus' deutlich höher [1].

eines jeden Blocks angibt:

$$\text{EME} = \frac{2}{k_1 k_2} \sum_{l=1}^{k_1} \sum_{k=1}^{k_2} \log \left(\frac{I_{\max,k,l}}{I_{\min,k,l}} \right) \quad (8)$$

$$\text{UISM} = \sum_{c=1}^3 \lambda_c \text{EME}(\text{grayscale edge}_c) \quad (9)$$

$(I_{\max,k,l})/(I_{\min,k,l})$ gibt das relative Kontrastverhältnis innerhalb eines jeden Blocks an und die EME-Werte (*enhancement measure estimation*) werden entsprechend der relativen visuellen Antwort auf den Rot-, Grün- und Blaukanal [1] mit folgenden Koeffizienten λ_c linear kombiniert: $\lambda_R = 0,299$, $\lambda_G = 0,587$, $\lambda_B = 0,114$. Leider ließ sich die Blockgröße, die Panetta et al. [1] verwenden, nicht herausfinden.

Abbildung 3 verdeutlicht, dass der UISM-Wert für Fotos mit größerer Schärfe höher ist.

Underwater image contrast measure (UIConM) Häufig ist Rückstreuung der Grund für den Kontrastverlust bei Unterwasserfotos. Deshalb nutzen Panetta et al. [1] die logAMEE-Messung [3], sodass der UIConM-Wert wie folgt berechnet werden kann:



UIConM = 0.4732



UIConM = 1.3237

Abbildung 4: (a) Originales Unterwasserfoto. (b) Unterwasserfoto nach Kontrastkorrektur. Der UIConM-Wert (siehe Formel (10)) ist nach Anwendung eines Korrektur-Algorithmus' deutlich höher [1].

$$\text{UIConM} = \text{logAMEE}(\text{Intensity}) \quad (10)$$

$$\text{logAMEE} = \frac{1}{k_1 k_2} \otimes \sum_{l=1}^{k_1} \sum_{k=1}^{k_2} \frac{I_{\max,k,l} \Theta I_{\min,k,l}}{I_{\max,k,l} \oplus I_{\min,k,l}} \times \log \left(\frac{I_{\max,k,l} \Theta I_{\min,k,l}}{I_{\max,k,l} \oplus I_{\min,k,l}} \right) \quad (11)$$

Wie beim UISM-Wert wird das Bild in $k_1 k_2$ Blöcke aufgeteilt. Die Symbole \otimes , \oplus und Θ sind sogenannte PLIP-Operationen (*parameterized logarithmic image processing*) [3]. Diese werden genutzt, da lineare Bildoperationen zwei wesentliche Nachteile haben [3]. Zum einen werden Intensitätswerte, die außerhalb des definierten Bereichs liegen, einfach abgeschnitten, was einen Verlust von Information bedeutet. Zum anderen stimmen lineare Operationen häufig nicht mit der visuellen menschlichen Wahrnehmung überein. Um mithilfe der PLIP-Operationen optimale Ergebnisse zu erreichen, müssen anwendungsspezifische Parameter trainiert werden. Diese waren in der Publikation von Panetta et al. [1] ebenfalls nicht erwähnt.

Auch für den UIConM-Wert gilt, dass Bilder, die in Bezug auf den Kontrast eine bessere Qualität aufweisen, einen höheren Wert zugewiesen bekommen (vgl. Abbildung 4).

Underwater image quality measure (UIQM) Zuletzt werden die drei Terme UICM, UISM und UIConM in der UIQM zusammengeführt [1]:

$$\text{UIQM} = c_1 \cdot \text{UICM} + c_2 \cdot \text{UISM} + c_3 \cdot \text{UIConM} \quad (12)$$

Die Parameter c_1 , c_2 und c_3 sind anwendungsabhängig. Sie geben die Möglichkeit, die einzelnen Attribute unterschiedlich stark zu gewichten. Panetta et al. [1] wählten als Standardparameter $c_1 = 0,0282$, $c_2 = 0,2953$ und $c_3 = 3,5753$, die sie aus einer multiplen linearen Regression mit einem Trainingsdatensatz von 30 mit unterschiedlichen Kameramodellen aufgenommenen Bildern aus unterschiedlichen Wassertiefen gewonnen haben.

Ein höherer UIQM-Wert entspricht einem qualitativ besseren Bild. UIQM-Änderungen von 10% sollen bereits für das menschliche Auge erkennbar sein [1].

2.2. Implementierung des Algorithmus' mittels C++ und OpenCV

Der in Kapitel 2.1 beschriebene Algorithmus wurde zunächst in C++ implementiert. Dabei wurde für grundlegende Bildoperationen auf die OpenCV-Bibliothek [4] zurückgegriffen. Der modulare Aufbau des Algorithmus, d.h. die Aufteilung in UICM, UISM und UIConM, wurde im Quellcode durch separate Klassen für die einzelnen Terme berücksichtigt, die schließlich in einer Hauptklasse zusammengeführt werden.

2.3. Integration in Copria

Copria (collaborative pattern recognition & image analysis) ist eine webbasierte Datenfluss-Prozessierungssoftware [5]. Das Ziel von Copria ist, eine effiziente Nutzung von komplexen Datenprozessierungs-Pipelines zu ermöglichen, auch in Zusammenarbeit mit mehreren Wissenschaftlern. Eine Pipeline besteht aus sogenannten Knoten, die grundlegende Operationen aus der Mustererkennung, Data-Mining oder Bildverarbeitung umsetzen. Diese Pipelines werden, um eine hohe Effizienz zu garantieren, auf dem *compute cluster* des CeBiTec ausgeführt.

Neben der Implementierung als eigenständiges C++-Programm wurde der in Kapitel 2.1 beschriebene Algorithmus auch in Copria integriert. Dazu wurden die drei Terme UICM, UISM und UIConM jeweils als eigenständige Knoten entwickelt. Diese können nach Belieben in einer Pipeline zusammengeführt werden, um den UIQM zu erhalten. Auch die Integration in andere Pipelines, wie z.B. zur Prüfung von Bildverbessernden Algorithmen, ist denkbar.

3. Ergebnisse und Diskussion

3.1. Vergleich der Ergebnisse des selbst implementierten Algorithmus' mit denen des Vorgegebenen

In der Publikation von Panetta et al. [1] ist Einiges nicht eindeutig beschrieben. So sind beispielsweise keine PLIP-Parameter angegeben und die Berechnung der Varianz für den UICM-Wert (siehe Formel (7)) ist eindeutig. Aus diesem Grund empfiehlt sich eine Evaluation des implementierten Algorithmus'. Dazu wurden Bilder aus [1] verwendet, für die Werte für alle vier Terme (UICM, UISM, UIConM, UIQM) angegeben waren. Die entsprechenden Bilder, 13c-i (vgl. Tabelle 6 in [1]), wurden, da sie nicht als Bilddateien vorlagen, aus dem PDF ausgeschnitten, um sie dem Programm zur Bewertung zu übergeben.

Im Folgenden werden zunächst die einzelnen Attribute und zuletzt auch der Gesamtterm evaluiert.

UICM Die vom implementierten Algorithmus bestimmten Werte stimmen sehr gut mit den in [1] angegebenen Werten überein (vgl. Abbildung 5). Die hohe Korrelation wird durch ein Bestimmtheitsmaß von 0,9888 bestätigt.

Die geringfügigen Abweichungen liegen möglicherweise in der unterschiedlichen Auflösungen der

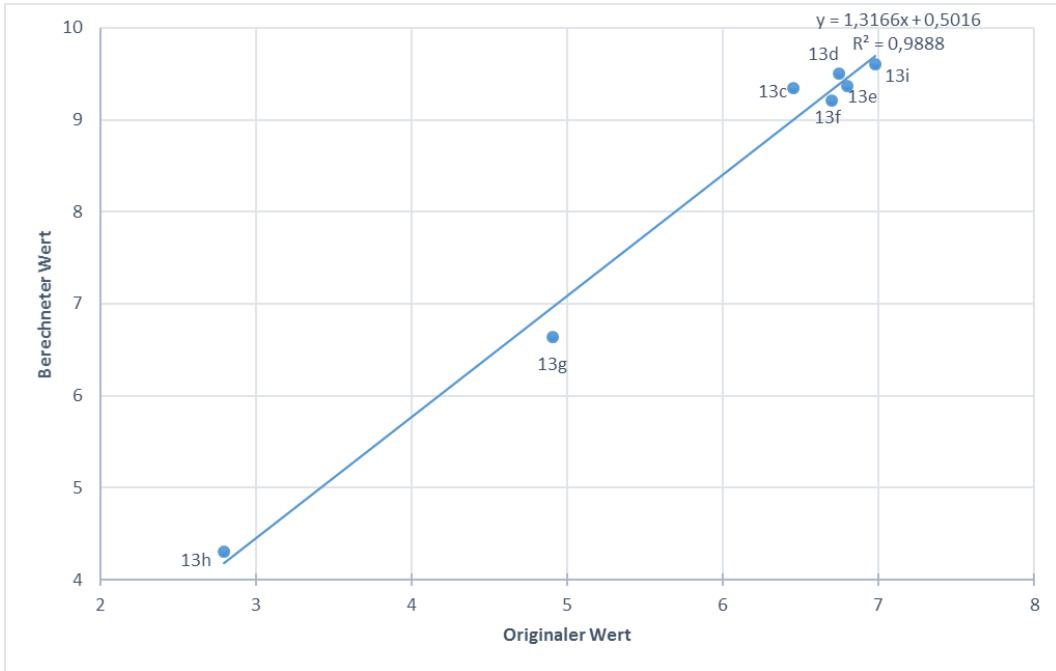


Abbildung 5: Vergleich der UICM-Werte des selbst implementierten Algorithmus' mit denen des Vorgegebenen für die Bilder 13c-i aus [1]. Auf der x-Achse ist der in [1] angegebene UICM-Wert aufgetragen, auf der y-Achse der UICM-Wert des implementierten Algorithmus'. Zudem sind die Funktion (y) der automatisch bestimmten Trendlinie sowie das Bestimmtheitsmaß (R^2) angegeben.

Bilddateien begründet. Zudem war unklar, ob die Varianz (vgl. Formel (5)) wie der Mittelwert (vgl. Formel (3)) nur zwischen T_{α_L} und T_{α_R} berechnet wird.

UISM Für die UISM-Messung liegt das Bestimmtheitsmaß deutlich geringer, nämlich bei $R^2 = 0,8303$ (vgl. Abbildung 6). Die Gründe dafür liegen sowohl in der wahrscheinlich unterschiedlichen Auflösung als auch an den möglicherweise verschiedenen Blockgrößen, da für die in [1] angegebenen Werte keine verbindliche Blockgröße genannt ist. Außerdem ist nicht erwähnt, wie bei einem minimalen Intensitätswert $I_{\min,k,l}$ von 0 vorgegangen werden muss, da durch $I_{\min,k,l}$ geteilt wird (vgl. Formel (9)).

In den Daten ist eine deutliche Korrelation erkennbar.

UIConM Der Werte zur Messung des Kontrasts zeigen ebenfalls eine deutliche Korrelation, wenn auch negativ (vgl. Abbildung 7). Das Bestimmtheitsmaß liegt mit $R^2 = 0,8935$ zwischen denen von UICM und UISM.

Bei der Implementierung dieses Terms traten vergleichsweise viele Unklarheiten auf. Zunächst war ebenfalls die Größe der Blöcke nicht angegeben. Daher wurde standardmäßig wie auch für UISM die in anderen Publikationen erwähnte Blockgröße von 4 gewählt. Des Weiteren war unklar, wie die PLIP-Parameter gewählt wurden. Da andere Veröffentlichungen (z.B. [3]) Werte von 1026 empfehlen, wurden diese standardmäßig für $\mu(M)$, $\gamma(M)$, $k(M)$ und $\lambda(M)$ übernommen. β wurde standardmäßig auf 1 gesetzt. Außerdem blieb bis zuletzt unklar, ob mit der Multiplikation \times in Formel (10) wie in [6] eine PLIP-Operation gemeint ist oder nicht.

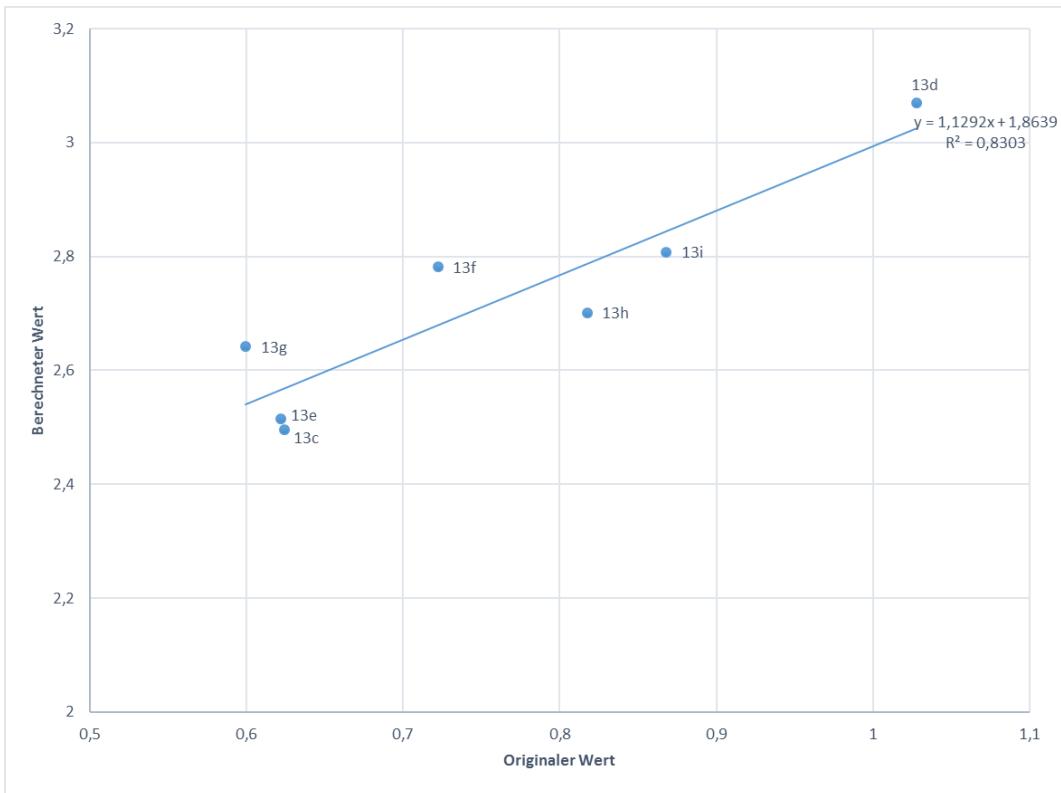


Abbildung 6: Vergleich der UISM-Werte des selbst implementierten Algorithmus' mit denen des Vorgegebenen für die Bilder 13c-i aus [1]. Auf der x-Achse ist der in [1] angegebene UISM-Wert aufgetragen, auf der y-Achse der UISM-Wert des implementierten Algorithmus'. Zudem sind die Funktion (y) der automatisch bestimmten Trendlinie sowie das Bestimmtheitsmaß (R^2) angegeben.

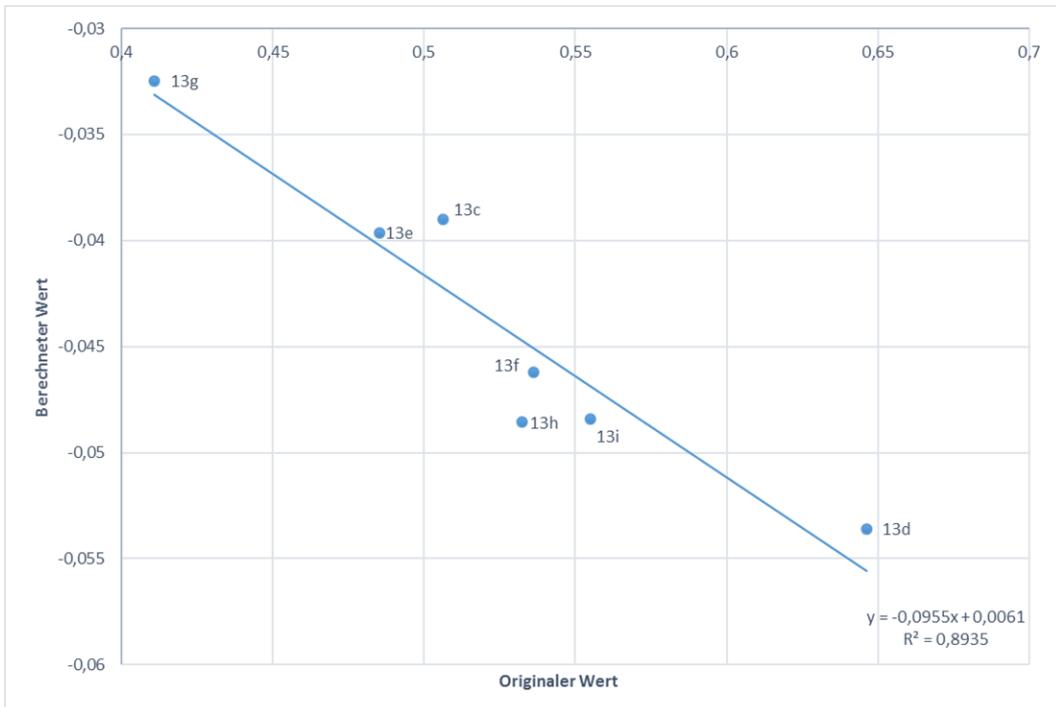


Abbildung 7: Vergleich der UIConM-Werte des selbst implementierten Algorithmus' mit denen des Vorgegebenen für die Bilder 13c-i aus [1]. Auf der x-Achse ist der in [1] angegebene UIConM-Wert aufgetragen, auf der y-Achse der UIConM-Wert des implementierten Algorithmus'. Zudem sind die Funktion (y) der automatisch bestimmten Trendlinie sowie das Bestimmtheitsmaß (R^2) angegeben.

Trotz der Unklarheiten liefert der implementierte Algorithmus ausreichend verlässliche Daten.

UIQM Da die Werte für die drei Terme UICM, UISM und UIConM des implementierten Algorithmus' nicht zu 100% mit den Daten aus [1] übereinstimmen, müssen die Konstanten aus Formel (12) mit der Steigung der jeweiligen Regressionsgeraden multipliziert werden. Die neuen Konstanten sind $c_1 = 0,03712812$, $c_2 = 0,33345$ und $c_3 = -0,34144$. Mithilfe dieser neuen Konstanten kann der Gesamtterm UIQM bestimmt werden.

Abbildung 8 zeigt, inwiefern die UIQM-Werte des implementierten Algorithmus' mit den Werten aus [1] korrelieren. Offensichtlich ist die Korrelation mit einem Bestimmtheitsmaß von $R^2 = 0,538$ deutlich schlechter als bei den einzelnen Termen. Der Wert für das Bild 13h sticht deutlich als Ausreißer hervor. Wird dieser Wert außen vor gelassen, liegt das Bestimmtheitsmaß bei $R^2 = 0,9383$. Möglicherweise ist in dem Paper an dieser Stelle ein Tippfehler vorhanden¹. Abschließend kann der Grund für diesen Ausreißer leider nicht bestimmt werden. Das nächste Unterkapitel wird allerdings zeigen, dass die Werte des implementierten Algorithmus gut mit der visuellen Wahrnehmung übereinstimmen. Trotzdem wäre eine weitergehende Überprüfung sinnvoll, allerdings ist sie mit den gegebenen Daten nicht umsetzbar.

¹Offensichtlich haben sich in die Tabelle VI in [1] mindestens zwei offensichtliche Fehler eingeschlichen. Zum einen handelt es sich bei den Werten um Werte für die Abbildung 13 und nicht 10 und zum anderen ist der UICM-Wert für 13c falsch. Statt 9,4518 ist ein Wert von 6,4518, wie auch in Abbildung 5 angenommen, deutlich plausibler, da der höchste Wert jeweils fett hervorgehoben ist. Zudem ergeben sich die UIQM-Werte nicht durch Multiplikation der einzelnen Terme mit den zugehörigen angegebenen Konstanten.

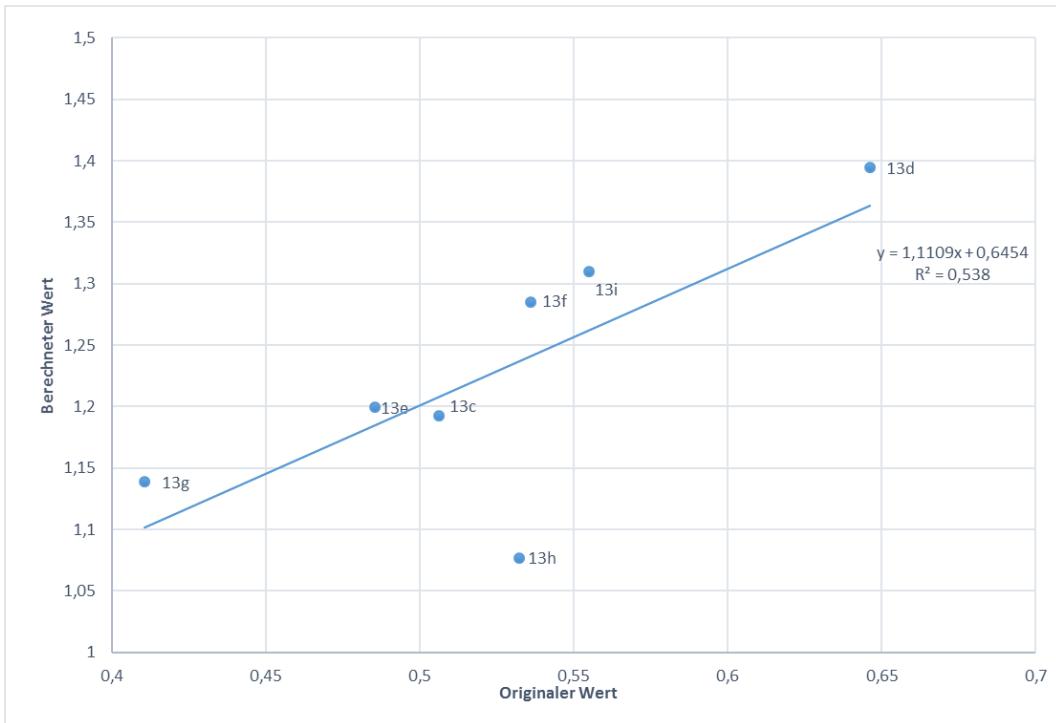


Abbildung 8: Vergleich der UIQM-Werte des selbst implementierten Algorithmus' mit denen des Vorgegebenen für die Bilder 13c-i aus [1]. Im Diagramm sind die in [1] angegebenen UIQM-Werte (x-Achse) gegen die UIQM-Werte des implementierten Algorithmus' (y-Achse) aufgetragen. Zudem sind die Funktion (y) der automatisch bestimmten Trendlinie sowie das Bestimmtheitsmaß (R^2) angegeben.

3.2. Evaluation des Algorithmus' mithilfe manuell sortierter Unterwasserfotos

Der Algorithmus wurde des Weiteren einer Überprüfung mit 20 Bildern von der Tiefsee-Station HAUSGARTEN unterzogen. Diese wurden zuvor manuell qualitativ absteigend sortiert. Anschließend wurden sie dem implementierten Programm übergeben und eine Rangkorrelation (Quadrat der Spearman-Korrelation) durchgeführt (vgl. Abbildung 9). Auch hier zeigt sich mit einem Bestimmtheitsmaß von fast 90% eine ausgeprägte Korrelation. Da die Unterschiede zwischen den Bildern teilweise sehr geringfügig sind, erscheint auch die vorgeschlagene Reihenfolge aufgrund der UIQM-Werte akzeptabel (vgl. Abbildung 10).

3.3. Verbesserungs- und Anpassungsmöglichkeiten

Die Evaluation des Programms hat offen gelegt, dass es an einer oder anderen Stelle noch Optimierungsbedarf gibt. Insbesondere die vergleichsweise geringe Korrelation der UIQM-Werte wirft Fragen auf. Da im Paper [1] einige Parameter nicht genannt werden, lassen diese Spielraum für Optimierungen. So könnte die Blöckgröße für sowohl UIISM als auch UIConM vor allem für hochauflösende Bilder erhöht werden. Veränderungen der Blockgröße haben nämlich signifikanten Einfluss auf die Werte. Zudem sind Untersuchungen zur Stabilität des Algorithmus' bei Änderungen der (PLIP-)Parameter oder des Eingabebildes (Bildgröße oder anderer Wertebereich der Farbwerte) denkbar.

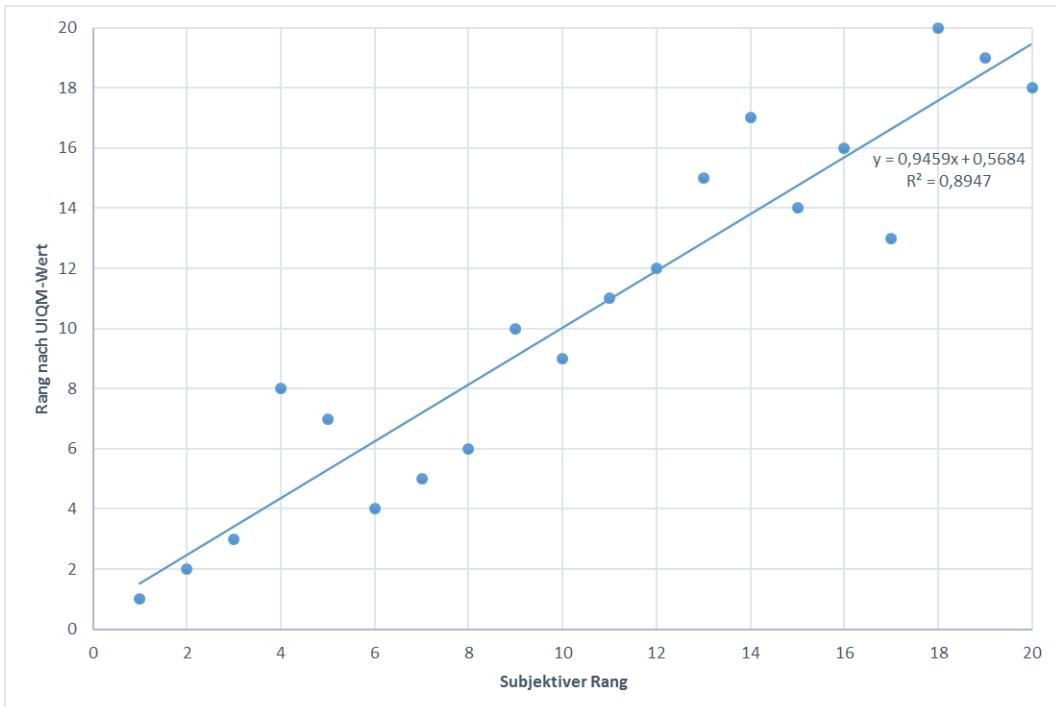


Abbildung 9: 20 Unterwasserfotos von der Tiefsee-Station HAUSGARTEN wurden sowohl einer subjektiven Qualitätsbewertung als auch einer Bewertung durch den UIQM-Algorithmus unterzogen und qualitativ absteigend sortiert. Das Diagramm zeigt die Rangkorrelation zwischen den subjektiven Rängen (x-Achse) und den Rängen nach UIQM-Werten (y-Achse). Zudem sind die Funktion (y) der automatisch bestimmten Trendlinie sowie das Bestimmtheitsmaß (R^2) angegeben.

4. Fazit

Dieses Projekt hat gezeigt, dass die von Panetta et al. [1] veröffentlichte Messmethode zur Bestimmung der Qualität von Unterwasserfotos prinzipiell gut mit der menschlichen visuellen Wahrnehmung korreliert. Das selbst implementierte Programm zeigt eine hohe Übereinstimmung mit den Werten aus dem Paper, auch wenn es einzelne Ausreißer gab, die nicht endgültig aufgeklärt werden konnten. Auch die Überprüfung mit selbst sortierten Unterwasserfotos zeigte eine überzeugende Übereinstimmung, sodass dieses Programm eine echte Hilfe für eine erste Einschätzung der Bildqualität sein kann. Da die einzelnen Teilterme UICM, UISM und UIConM nun auch von Copria-Knoten berechnet werden, können sie einfach in Copria-Pipelines integriert werden.

Trotz der grundsätzlich positiven Ergebnisse bleibt anzumerken, dass die Umsetzung des Algorithmus' aufgrund von Unklarheiten und Fehlern in der Veröffentlichung erheblich erschwert wurde. Aus dem Grund sind weitere Tests und Anpassungen sinnvoll.

Literatur

- [1] Karen Panetta, Chen Gao, and Sos Agaian. Human-Visual-System-Inspired Underwater Image Quality Measures. *IEEE Journal of Oceanic Engineering*, PP(99):1–11, 2015.
- [2] J.S. Jaffe, Silvia Corchs, B McGlamery, JS Jaffe, et al. Computer modeling and the design of optimal underwater imaging systems. *IEEE Journal of Oceanic Engineering*, 15(2):101–111, apr 1990.
- [3] Karen Panetta, Sos Agaian, Yicong Zhou, and Eric J Wharton. Parameterized Logarithmic Framework for Image Enhancement. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(2):460–473, apr 2011.
- [4] Gary Bradski. The opencv library. *Doctor Dobbs Journal*, 25(11):120–126, 2000.
- [5] Timm Schoening. Timm Schoening - Current projects. <http://timmschoening.de/www/index.php?p=projects>.
- [6] Eric Wharton, Sos Agaian, and Karen Panetta. A logarithmic measure of image enhancement. pages 62500P–62500P–12. International Society for Optics and Photonics, may 2006.
- [7] Michel Jourlin and Jean-Charles Pinoli. A model for logarithmic image processing. *Journal of Microscopy*, 149(1):21–35, jan 1988.

A. Anhang

A.1. Fotos der Tiefseestation HAUSGARTEN zur Überprüfung des Algorithmus'

M	P	UIQM	M	P	UIQM
1	1	1,1219	6	4	1,0290
2	2	1,1027	7	5	0,9823
3	3	1,0643	8	6	0,9542
4	8	0,9121	9	10	0,8057
5	7	0,9447	10	9	0,8498

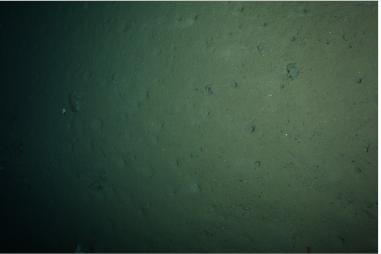
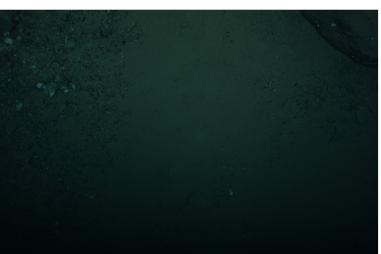
M	P	UIQM	M	P	UIQM		
	11	11	0,8002		16	16	0,0,7465
	12	12	0,7950		17	13	0,7883
	13	15	0,7582		18	20	0,7167
	14	17	0,7405		19	19	0,7211
	15	14	0,7599		20	18	0,7260

Abbildung 10: In der Abbildung sind die 20 ausgewählten Fotos der Tiefsee-Station HAUSGARTEN in der manuell festgelegten Reihenfolge (qualitativ absteigend) sortiert (Spalte M). Des Weiteren ist der Rang nach UIQM-Werten (Spalte P) sowie der UIQM-Wert selbst (Spalte UIQM) angegeben.