UNIVERSITÄT KOBLENZ · LANDAU
Fachbereich 4: Informatik

AGAS
Arbeitsgruppe Aktives Sehen

Feature Detection und Matching Verfahren zur Position- und Lagebestimmung

Diplomarbeit

zur Erlangung des Grades eines/r

Diplom-Informatikers / Diplom-Informatikerin
im Studiengang Computervisualistik

vorgelegt von

Jean-Claude Rosenthal

Betreuer:	Prof. Dr.-Ing. Dietrich Paulus, Institut für Computervisualistik, Fachbereich Informatik
Erstgutachter:	Prof. Dr.-Ing. Dietrich Paulus, Institut für Computervisualistik, Fachbereich Informatik
Zweitgutachter:	Dipl.-Inf. Johannes Pellenz, Institut für Computervisualistik, Fachbereich Informatik

Koblenz, im Oktober 2006

Erklärung

Ich versichere, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Die Richtlinien der Arbeitsgruppe für Studien- und Diplomarbeiten habe ich gelesen und anerkannt, insbesondere die Regelung des Nutzungsrechts

Mit der Einstellung dieser Arbeit in die Bibliothek bin ich einverstanden. ja ☐ nein ☐

Der Veröffentlichung dieser Arbeit im Internet stimme ich zu. ja ☐ nein ☐

Koblenz, den ..... Unterschrift

Zusammenfassung

Mit Hilfe von Stereobildfolgen, die ein Stereokamerasystem liefert, wird versucht Informationen aus der betrachtenden Szene zu gewinnen. Die Zuordnung von Bildpunkten, die in beiden Bildern eines Stereobildpaares vorkommen und einen gemeinsamen Weltpunkt beschreiben, ermöglichen die Bestimmung einer Tiefeninformation. Das Extrahieren von Bildpunkten und deren Zuordnung sind die entscheidenen Faktoren zur Gewinnung der Tiefeninformation. Die Tiefe erlaubt es Aussagen über die Struktur der aufgenommenen Szene zu machen. Bei Übertragung dieser Idee auf das Verfolgen von gemeinsamen Weltpunkten in Bildsequenzen ist es möglich eine relative Positions- und Lageschätzung des Kamerasystems zur vorher aktuellen Position zu bestimmen. Schwierigkeiten ergeben sich aus Verdeckungen von Weltpunkten für den jeweiligen Sensor, sowie fehlerhaften Bildpunktzuordnungen. Die Geschwindigkeit des kombinierten Vorgang aus Extraktion und Punktzuordnung stellt eine weitere Anforderung an das System.

Inhaltsverzeichnis

1 Einleitung **5**

2 Struktur in Stereobildfolgen **9**

2.1 Stereosehen mit Kameras 9

2.1.1 Kameramodell 10

2.1.2 Kalibrierung 13

2.1.3 Ideales Kameramodell 16

2.1.4 Rektifizierung 17

2.2 Feature Extraktoren 19

2.2.1 Moravec-Operator 19

2.2.2 Kanade - Lucas - Tomasi Operator (KLT) 20

2.2.3 Harris Corner - Operator 21

2.2.4 SIFT - Operator 22

2.3 Matching Metriken 27

2.3.1 Lokale Korrespondenzanalyse 28

2.3.2 Globale Korrespondenzanalyse 33

3 Intra- und Inter-Matching von Punktmengen **39**

3.1 Definition: Intra-Matching und Inter-Matching 40
3.2 Klassisches Tracking mit Intra- / Inter-Matching 43
3.3 Multiresolution Tracking mit Intra- / Inter-Matching 44
3.4 SIFT-Operator und Intra- / Inter-Matching 47
3.5 Erweiterung zu Intra- / Inter-Rematching 48
3.6 Abgrenzung zu anderen Arbeiten 48

4 Experimente und Ergebnisse **51**
4.1 Versuchsaufbau 52
4.1.1 Reales Stereosystem 52
4.1.2 Synthetisches Stereosystem 52
4.2 Versuchsdurchführung 54
4.3 Ergebnisse 54
4.3.1 Feature-Entwicklung 55
4.3.2 Lage- und Positionsschätzung 58
4.4 Validierung 73

5 Zusammenfassung **77**

A Verschiedenes **81**

B Danksagung **85**

Kapitel 1

Einleitung

In dieser Diplomarbeit wird das Ziel verfolgt schnelle Feature Detection- und Matching Algorithmen zum Tracking von Punktmengen zu testen. In Zusammenarbeit mit dem Deutschen Zentrum für Luft- und Raumfahrt (DLR) in Berlin - Adlershof werden hier Verfahren vorgestellt, die eine schnelle und robuste relative Position- und Lagebestimmung eines Stereokamerasystems aus bestätigten Punktkorrespondenzen in Räumen ermöglichen sollen. Daraus kann das System messen wie stark eine Bewegung von Zeitpunkt t_0 zum Zeitpunkt t_1 ausfällt. Das Anwendungsszenario soll dabei die Navigation von autonomen Robotern in Innenräumen sein, wobei eine Erweiterung auf beliebige Anwendungsszenarien nicht ausgeschlossen ist. Die Arbeit versucht dabei eine Erklärung zu liefern was ein gutes Feature ausmacht und inwieweit sich diese in Hinblick auf Robustheit und Schnelligkeit mit o. g. Anwendungsszenario vereinbaren lassen. Weiterhin werden verschiedene Strategien beim Matching untersucht, die u. a. versuchen die Genauigkeit der als korrekt betrachtenden Zuordnungen von Features zu erhöhen um somit fehlerhafte Korrespondenzen auszuschließen.

Im Gegensatz dazu stellt die angestrebte Schnelligkeit des Verfahrens eine weitere Anforderung an das System. So lässt bereits die Aufgabenstellung erahnen, dass die angestrebten Ziele, Robustheit und Schnelligkeit, im Widerspruch zueinander stehen werden. Die beiden wichtigsten Ziele des Trackings führen somit zu einer gegenseitigen Beeinflussung des gewünschten Ergebnisses.

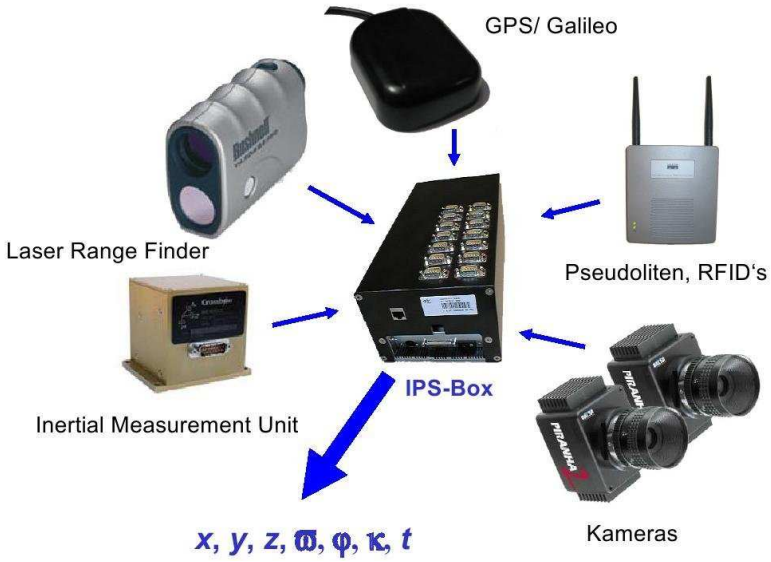


Bild 1.1: Ein möglicher multisensorieller Systemmessaufbau zur Bestimmung der gewünschten Parameter von Position (x, y, z), Lage (ϖ,φ,κ) und Zeit(t)

Oft ist es jedoch so, dass ein solches System aus mehr als nur einer Sensorkomponente besteht. In der Regel werden mehrere Sensorsysteme zeitgleich eingesetzt, die das Ergebnis des jeweils anderen Sensor stützen können und sollen, um die Selbstlokalisation zu verbessern. Der multisensorielle Ansatz bietet den Vorteil, dass die Schwächen, die jeder Sensor, unter bestimmten Bedingungen mit sich bringt, durch andere Messeinheiten ausgeglichen werden können. Als Sensoren werden, um nur einige zu nennen, u. a. Kamerasysteme, sowohl stereobasierte als auch monokulare Aufbauten, Neigungssensoren, inertiale Messeinheiten, Zugangspunkte (Pseudoliten) oder auch GPS eingesetzt. Bild 1.1 zeigt eine Variante eines möglichen Aufbaus. Zur Positions- und Lagebestimmung werden dann die jeweiligen Sensordaten miteinander fusioniert. Der in [WB01] beschriebene Kalmanfilter dient zur Gewichtung der Messdaten. Die Zustandsschätzung ergibt sich dann aus den gewichteten Messdaten und erfolgt durch die Berechnung der sieben zu bestimmenden Größen: Position (x, y, z), Lage (ϖ,φ,κ) in Relation zur jeweiligen Zeit(t).

Im Zusammenhang mit der Aufgabenstellung sind aktuelle, relevante Arbeiten, die sich mit der Problematik der Merkmalsextraktion befassen z. B. hier zu finden [SMB00], [MS04],

[Low04]. Das Problem des Stereo-Matching von Punkten wird in folgenden Arbeiten genauer untersucht [BVZ01], [KZ01], [Hir03], [SSZ01], [BBH03]. Für das gezielte Verfolgen von korrepondierenden Punktmengen finden sich in [NBN06], [Hir03] und [ZDFL95] praxisnahe Umsetzungen.

Der Aufbau der Arbeit ist wie folgt: Kapitel 2 gibt einen Überblick über vorhandene Ansätze von Feature-Detection und Feature-Matching Algorithmen, die unter verschiedenen Bedingungen angewendet werden können in Abhängigkeit zum Anwendungsszenario. In Kapitel 3 wird ein Tracker vorgestellt, der verschiedene Verfahren aus Kapitel 2 miteinander kombiniert. Weiterhin wird eine bestimmte Strategie beim Tracking vorgestellt. In Kapitel 4 werden die erzielten Ergebnisse der verschiedenen Kombinationen aus Extraktoren und Matching-Algorithmen ausgewertet. Eine Zusammenfassung der Ergebnisse erfolgt in Kapitel 5. Dort werden aus den resultierenden Erkenntnissen mögliche Weiterentwicklungen und Vorgehensweisen für das Tracking in geschlossenen und nicht geschlossenen Räumen diskutiert.

Kapitel 2

Struktur in Stereobildfolgen

In Abschnitt 2.1 wird der generelle Aufbau eines Stereokamerasystems beschrieben und welche Problematik und Möglichkeiten ein solcher Aufbau bietet. In Abschnitt 2.2 werden Feature Extraktoren vorgestellt und beschrieben. In Abschnitt 2.3 werden ausgewählte Matching Metriken und deren Vor- und Nachteile erläutert. Dabei erfolgt eine Unterteilung in globale und lokale Verfahren zur Korrespondenzbestimmung.

2.1 Stereosehen mit Kameras

Das Stereosehen ermöglicht den Menschen erst das drei-dimensionale Sehen. Durch die Bestimmung des Schnittpunktes der beiden Sehstrahlen erhält der betrachtete Punkt bzw. die komplette Szene eine Tiefe und ermöglicht somit die Koordination im dreidimensionalen Raum. Diese Idee versucht man nun auf technische Geräte zu übertragen. Die Umsetzung erfolgt dabei mit zwei auf dieselbe Szene ausgerichteten Kameras, die auf einer gemeinsamen Achse liegen. Man versucht die Ausrichtung der Kameras so parallel wie möglich zu gestalten. Dies jedoch zu erreichen und somit absolut achsenparallele Sichtstrahlen zu erhalten, stellt eine Schwierigkeit dar, die bereits durch die unterschiedliche technische Beschaffenheit der Kameras verursacht wird. Die Ausrichtungsgenauigkeit der Kameras zueinander stellt ein Problem dar, das in Abhängigkeit zur Umgebung und der gewünschten Anwendung steht, so dass die gewünschte Achsenparallelität in der Regel

nicht erreicht werden kann.

Der typische Aufbau eines Stereokamerasystems wird in Bild 2.1 gezeigt, welcher die geometrischen Zusammenhänge verdeutlicht. Ziel ist es mittels Triangulierung der Sehstrahlen für die Bildpunkte p^c, q^c, des jeweiligen Kamerasystems, den gemeinsamen Weltpunkt p^w in 3D - Koordinaten zu berechnen und somit die zusätzliche Tiefeninformation zu erhalten.

Die Zusammenhänge die hier dargestellt sind, beschreiben die so genannte Epipolargeometrie [Pau03], [HZ03]. Die Epipolargeometrie lässt Aussagen über bestimmte Sachverhalte zu. So schneidet die Epipolarebene die beiden Bildebenen B_1 und B_2 auf einer Geraden. Daraus leitet sich Koplanaritätsbedingung ab. Ein gemeinsamer Weltpunkt p^w, der auf die Bildebene B_1 der linken Kamera abgebildet wird, entspricht also einer Linie im rechten Kamerabild I_2. Diese Linie heißt Epipolarlinie (l_1, l_2). Dabei gilt für alle epipolaren Linien, dass sie den Epipol schneiden. Die Epipole (e_1, e_2) sind die Schnittpunkte von $\boldsymbol{t}$ mit der jeweiligen Bildebene. Mittels $\boldsymbol{R}$ und $\boldsymbol{t}$ werden die Transformationsparameter beschrieben, die den Ursprung des einen Kamerakoordinatensystem in das jeweils andere transformieren.

Probleme, die sich aus dem Aufbau zur Bestimmung eines gemeinsamen Weltpunktes ergeben sind u. a. Verdeckungen, die durch den unterschiedlichen Aufnahmewinkel entstehen, ungleichmäßige Beleuchtung der Aufnahmeszene, verursacht z. B. durch Schatten, schnelle Bewegungen des Systems bei zu niedriger Aufnahmefrequenz, die eine fehlerhafte Korrespondenzbestimmung in aufeinanderfolgenden Frames verursacht, wegen nicht vorhandenen Überlappungsbereichen in der Szene. Dies alles sind Probleme, die das menschliche Sehen mehr oder weniger alleine lösen kann, die aber die technische Umsetzung vor große Probleme stellt.

2.1.1 Kameramodell

Als Modell wird das Lochkameramodell angenommen. Dieses Kameramodell besteht aus zwei Teilen, die durch bestimmte Parameter beschrieben werden. Zum einem sind das die extrinsischen Kameraparameter, die die Bewegungen, bestehend aus Rotationen $\boldsymbol{R}$ und Translationen $\boldsymbol{t}$, des Kamerakoordinatensystems (KKS) in Bezug zum Weltkoordi-

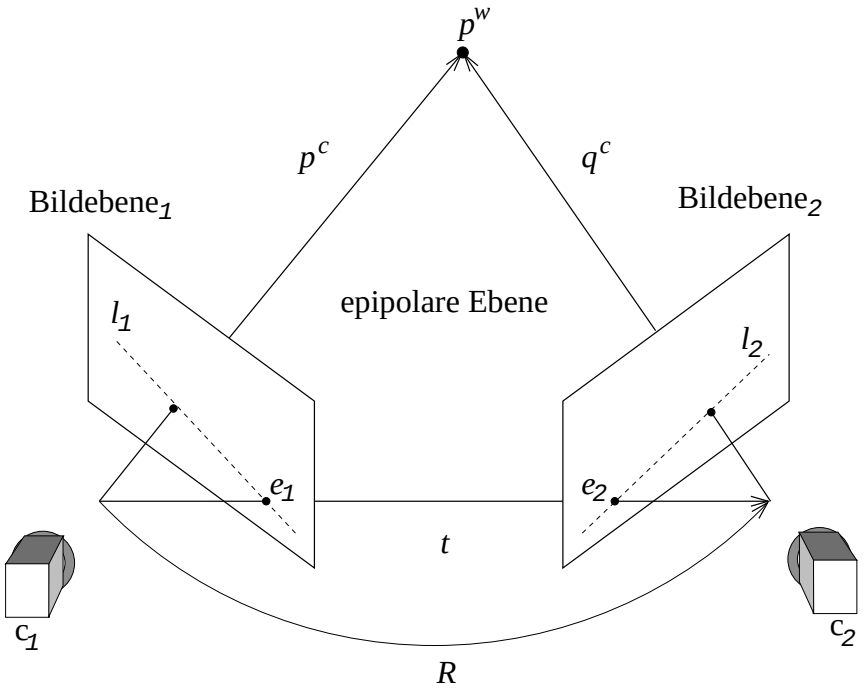


Bild 2.1: Epipolargeometrie: l_1, l_2 epipolare Linien, e_1, e_2 Epipole, c_1, c_2 Kamerasysteme, p^w Weltpunkt in 3D, p_1^c,q_2^c Kamerapunkte in 2D, $\boldsymbol{t}$, $\boldsymbol{R}$ Transformationsparameter

natensystem (WKS) beschreiben. Zum anderen sind das die intrinsischen Kameraparameter, die die Kameraeigenschaften, wie Brennweite (f_x, f_y), Hauptpunkt (H_x, H_y) und Pixelscherung (γ) beinhalten. Der Hauptpunkt repräsentiert die Stelle auf der Bildebene, an welcher die optische Achse diesen orthogonal dazu schneidet. Im Allgemeinen entspricht die Bildmitte nicht dem Hauptpunkt, sondern liegt häufig leicht versetzt davon. Die Scherung eines quadratischen Pixel wird bei CCD-Kameras allgemein als Null betrachtet und vernachlässigt. Aus der angenommen Quadratur des Pixel ergibt sich ebenfalls, dass $f_x = f_y$ ist.

Mit Hilfe dieser Parameter lässt sich der komplette Abbildungsvorgang eines gemeinsamen Weltpunktes wie in Gleichung 2.1 modellieren:

$$\text{Weltpunkt } p^w \mapsto \text{Kamerapunkt } p^c \mapsto \text{Bildpunkt } p^i \mapsto \text{Pixel } p^p \tag{2.1}$$

Gleichung 2.2 beschreibt die Zusammenhänge zwischen Weltpunkt p^w und Kamerapunkt

p^c.

$$p^c = \boldsymbol{R_l} * p^w + \boldsymbol{t} \tag{2.2}$$

Die Bestimmung der intrinsischen Parameter erlaubt die Korrektur der Linsenverzeichnung und die perspektivische Projektion der Punkte auf die Bildebene. Die Projektion erfolgt dabei durch die Division des z-Wertes mit dem Punkt p^c in Kamerakoordinaten auf die Bildebene mit einer Tiefe $z = 1$.

$$p^i = \begin{pmatrix} p_x^i \\ p_y^i \\ p_z^i \end{pmatrix} = \begin{pmatrix} p_x^c / p_z^c \\ p_y^c / p_z^c \\ 1 \end{pmatrix} \tag{2.3}$$

$$\begin{aligned} \hat{p}_x^i &= p_x^i + p_x^i * (\kappa_1 * ({p_x^i}^2 + {p_y^i}^2) + \kappa_2 * ({p_x^i}^2 + {p_y^i}^2)^2) \\ \hat{p}_y^i &= p_y^i + p_y^i * (\kappa_1 * ({p_x^i}^2 + {p_y^i}^2) + \kappa_2 * ({p_x^i}^2 + {p_y^i}^2)^2) \end{aligned} \tag{2.4}$$

Der entzerrte Punkt $\hat{p}^i$ wird dann mit der Projektionsmatrix $\boldsymbol{P}$ aus Gleichung 2.6 in homogene Pixelkoordinaten $\tilde{p}^p$ transformiert. s ist ein Skalierungsfaktor, der sicherstellt, dass die letzte Koordinate des homogenen Punktes $\tilde{p}^p$ gleich 1 ist.

$$s * \tilde{p}^p = \boldsymbol{P} * \hat{p}^i \tag{2.5}$$

mit

$$s * \tilde{p}_z^p = 1 \text{ , wenn } \hat{p}_z^i \neq 1 \quad \boldsymbol{P} = \begin{pmatrix} f_x & \gamma & H_x \\ 0 & f_y & H_y \\ 0 & 0 & 1 \end{pmatrix} \tag{2.6}$$

Erweitert man diese Abbildung von Weltkoordinaten zu Pixelkoordinaten auf eine zweite Kamera, so ergeben sich für das Modell 20 Parameter (6 extrinsische Parameter und 2×7 intrinsische Parameter), die es zu bestimmen gilt. Da die beiden Kameras fest zueinander bleiben, reicht es aus die 6 extrinsischen Parameter nur einmal zu bestimmen. Die Umrechnung lässt sich leicht bestimmen aus Gleichung 2.2 durch auflösen nach p^w und einsetzen in Gleichung 2.7 und ergibt somit das Ergebnis in Gleichung 2.8 für die intrinsischen Parameter des zweiten Punktes des Stereosystems.

$$q^c = \boldsymbol{R_r} * p^w + \boldsymbol{t} \tag{2.7}$$

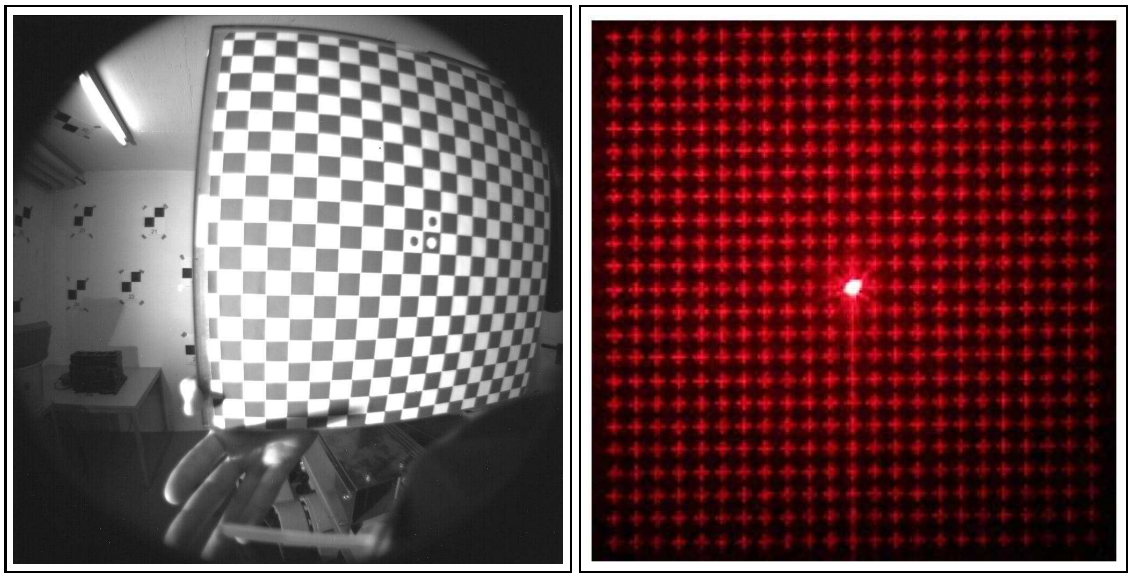


Bild 2.2: [v.l.n.r.] Kalibriermuster der linken Kamera; mit einem diffraktiven optischen Element erzeugtes Beugungsgitter.

$$\begin{aligned} q^c &= \boldsymbol{R_r}\boldsymbol{R_l}^{-1}(p^c - \boldsymbol{t_l}) + \boldsymbol{t_r} \\ &= \boldsymbol{R_r}\boldsymbol{R_l}^{-1}(p^c - \boldsymbol{t_l} + \boldsymbol{R_r}\boldsymbol{R_l}^{-1}\boldsymbol{t_r}) \\ &= \boldsymbol{R}(p^c - \boldsymbol{t}) \end{aligned} \tag{2.8}$$

2.1.2 Kalibrierung

Die Kalibrierung für das in Unterabschnitt 2.1.1 beschriebene Kameramodell ist ein entscheidender Schritt damit mit der Verarbeitung von Stereobildpaaren begonnen werden kann. Mit Hilfe der daraus gewonnenen Parameter lassen sich die entzerrten Bildkoordinaten in Gleichung 2.4 bestimmen. Von Zhang [Zha00] und Tsai [Tsa87] gibt es zwei angewandte Verfahren zur Kalibrierung von Kamerasystemen, die mit Hilfe eines Kalibriermusters korrigiert werden. Beide werden in der Praxis häufig eingesetzt. Ein solches Kalibriermuster zeigt Abbildung 2.2. In der Regel ist dabei die radiale Verzeichnung das größere Problem. Die Vernachlässigung der tangentialen Verzeichnung beeinflusst das Er-

Bild 2.3: Laboraufnahme: linke und rechte Bildinformation des Stereosystems

gebnis der Kalibrierung nicht gravierend [DF01]. Die beiden Kameras, die in diesem System eingesetzt werden haben eine sehr starke Verzeichnung. In Abbildung 2.3 ist ein solches Stereobildpaar zu sehen, dass vom System geliefert wird. Auffällig ist hierbei die besonders starke Verzeichnung der Radialebene. Aktuell werden beim DLR drei verschiedene Ansätze zur Kamerakalibrierung verfolgt. Aufgrund der extremen Korrekturbedürftigkeit der Bilder besteht zu Recht die Annahme, dass klassische Methoden zur Kalibrierung hier fehlschlagen werden.

Zum einen wird der softwarebasierte Ansatz verfolgt, d. h. das die Korrektur nur an Hand der vorliegenden Bildinformation bestimmt wird. Das vermessene Kalibriermuster gibt dabei die Referenz für die abzubildenden Bildpunkte an. Die Umsetzung dieser Variante ist zurzeit Gegenstand einer weiteren Diplomarbeit beim DLR. Die Kalibrierung wird mit einer modifizierten Version von Zhang arbeiten. Bei Abschluss dieser Arbeit lagen leider jedoch noch keine aussagefähigen Ergebnisse vor.

Ein zweiter Ansatz versucht mit Hilfe von so genannten diffraktiven optischen Elementen (DOE) – auch computergenerierte Hologramme genannt (CGH) – die Kamerageometrie zu bestimmen. DOEs werden allgemein zur Lichtmodulation verwendet. Im Sinne der

geometrischen Sensorkalibrierung werden DOE als Strahlteiler verwendet, dabei kommen die Effekte der Lichtbeugung zur Anwendung. An einer periodischen Mikrostruktur wird ein Beugungsgitter mit hochgenauen, bekannten Beugungswinkeln erzeugt, diese werden im Rahmen der Kalibrierung zur Bestimmung der Pixelblickrichtung verwendet. In dem Messverfahren wird ein Laserstrahl aufgeweitet, kollimiert auf das DOE gelenkt. Das Beugungsgitter wird auf der Fokalebene des zu kalibrierenden Sensors abgebildet. An Hand der bekannten Beugungswinkel kann die ideale Position einer Beugungsfigur auf der Bildebene ermittelt und mit der realen Position verglichen werden. Im Gegensatz zur herkömmlichen goniometrischen Messmethode können aufgrund der Strahlteilung viele Blickrichtungen gleichzeitig realisiert werden. Die Kalibrierung kann mit einem wesentlich geringeren Zeitaufwand für mehr Pixel durchgeführt werden als es z. B. im nachfolgenden Ansatz geschieht.

Eine dritte Methode nutzt einen Messaufbau mit einem Manipulator. Die CCD - Kamera kann nun exakt bezüglich einer bestimmten Pixelblickrichtung ausgerichtet werden. Das entsprechende Pixel wird dann mit einem Kollimator beleuchtet. Aus den am Manipulator eingestellten Winkeln kann analog zur zweiten Methode die ideale Koordinate bzw. der Durchstoßpunkt durch die Bildebene ermittelt und mit den gemessenen Koordinaten des beleuchteten Pixel verglichen werden.

Wird dies für eine ausreichend große Anzahl an Punkten durchgeführt, so lassen sich über einen Ausgleich die Verzeichnungsparameter, aus der Differenz zwischen der Messung und den vorgegebenen idealen Koordinaten, für das gesamte Bild daraus ableiten [SB00]. Diese sehr genaue Methode der Kalibrierung hat allerdings den Nachteil, dass sie sehr zeitintensiv bei hohen Kameraauflösungen ist, da jeder Pixel einzeln beleuchtet werden muss. Das hier beschriebene Verfahren wurde zur Kalibrierung der ADS 40 Kamera angewandt, „der ersten digitalen kommerziellen Luftbildkamera[1]“.

Nach einem Kalibrierungsprozess sind die 7 intrinsischen Kameraparameter aus Tabelle 2.1 bekannt.

[1] Andreas Eckardt, DLR-Berlin-Adlershof

κ_1	$\hat{=}$	1.Verzeichnungskoeffizient
κ_2	$\hat{=}$	2.Verzeichnungskoeffizient
γ	$\hat{=}$	Skew / Scherung
f_x	$\hat{=}$	hori. Brennweite
f_y	$\hat{=}$	vert. Brennweite
H_x, H_y	$\hat{=}$	Hauptpunkt der Kamera

Tabelle 2.1: zu bestimmende intrinsische Kameraparameter durch die Kalibrierung der Kamera.

2.1.3 Ideales Kameramodell

Nach erfolgreicher Kalibrierung und den nun bekannten inneren Parametern der Kamera, lässt sich das in Unterabschnitt 2.1.1 vorgestellte Kameramodell in ein Ideales transformieren. Ziel ist es nun die beiden Kamerakoordinatensysteme so auszurichten, dass folgende Bedingungen erfüllt sind

$$\boldsymbol{P} = \boldsymbol{P_l} = \boldsymbol{P_r} = \begin{pmatrix} f & 0 & \frac{w}{2} \\ 0 & f & \frac{h}{2} \\ 0 & 0 & 1 \end{pmatrix}, \ \boldsymbol{R} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \ \boldsymbol{t} = \begin{pmatrix} t \\ 0 \\ 0 \end{pmatrix} \quad (2.9)$$

Die Annahme dieses Modells bietet für den Stereoaufbau mehrere Vorteile. Zum einem ist nun sichergestellt, dass ein Weltpunkt p^w auf dieselbe Pixelzeile in beiden Bildern abgebildet wird. Die epipolare Bedingung garantiert, dass die epipolaren Linien parallel zu den Bildzeilen verlaufen wie in Abbildung 2.5 dargestellt. Außerdem gilt für einen Punkt p^p, dass sein korrespondierender Punkt q^p auf der epipolaren Linie mit der gleichen Bildspalte oder einer kleineren liegen muss. Die Differenz, die sich aus den unterschiedlichen x-Koordinaten ergibt, nennt man Disparität. Die Disparität fällt umso größer aus, je näher der jeweilige Punkt in der Szene an der Kamera liegt. Aus den Disparitäten lassen sich somit später Aussagen über die Tiefe eines gemeinsamen Weltpunktes treffen.

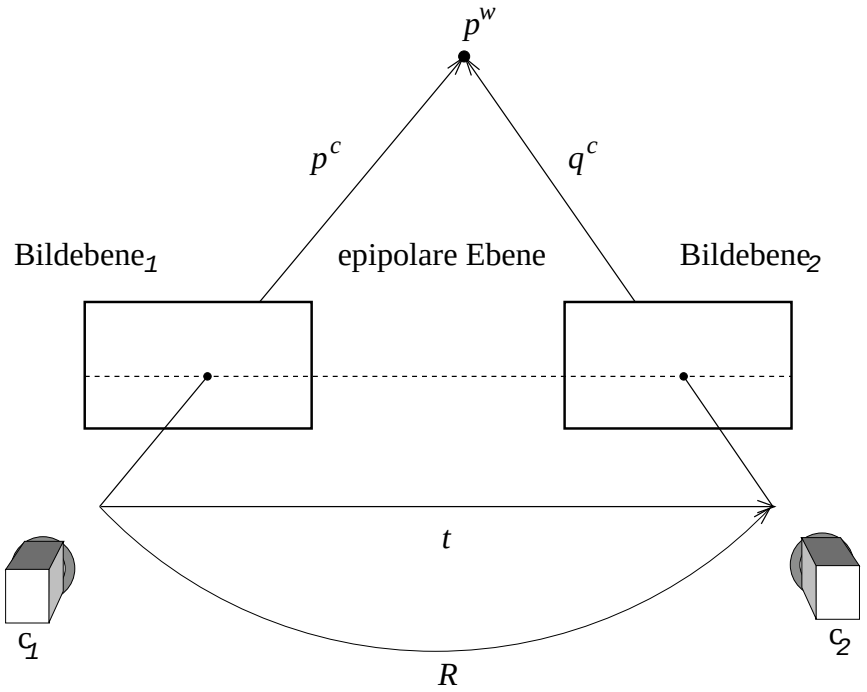


Bild 2.4: ideale, rektifizierte Geometrie: c_1, c_2 Kamerasystem, p^w Weltpunkt in 3D, p_1^c,q_2^c Kamerapunkte in 2D, $\boldsymbol{t}$, $\boldsymbol{R}$ Transformationsparameter

2.1.4 Rektifizierung

Die Transformation in das ideale Kameramodell entspricht einer Verschiebung der beiden Kamerakoordinatensysteme auf eine gemeinsame Ebene, so dass die epipolaren Linien parallel zu den Bildzeilen verlaufen. Punkte aus beiden Bildern liegen nun in einer gemeinsamen Bildebene, deren Abstand zu beiden optischen Zentren gleich ist. Um dies ohne größeren Informationsverlust zu erreichen, müssen zwei Bedingungen erfüllt sein.

Erstens wird die Entfernung zwischen der gemeinsamen Ebene und den beiden optischen Zentren festgelegt, die somit der neuen Brennweite entspricht. Zweitens muss die Ebene um die Linie, die durch die optischen Zentren geht, so rotiert werden, dass die gemeinsame Brennweite und Orientierung der Kameras nahe an deren eigenen Ausgangswerten liegt, um Interpolationsfehler zu vermeiden [GN01].

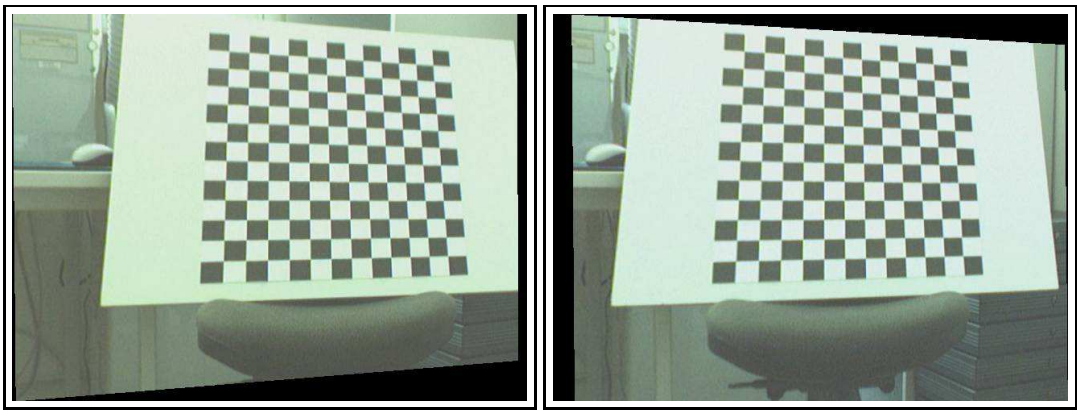


Bild 2.5: Beispiel für ein rektifiziertes Stereobildpaar. Dank an Peter Decker

Dies ermöglicht für die spätere Korrespondenzsuche die Einschränkung des Suchraumes auf eine 1D-Suche. Der korrespondierende Punkt muss also nur noch auf einer Linie gesucht werden (gegeben durch die epipolare Linie) und nicht mehr innerhalb eines Blocks. Zusätzlich wird auch noch der Suchbereich auf der Linie eingeschränkt. Eine genauere Beschreibung von Methoden zur Rektifizierung von Stereoaufbauten finden sich hier: [TV98], [Ora01], [FTV97]. Abbildung 2.5 zeigt ein solches rektifiziertes Bildpaar [2]. Den darin enthaltenen schwarzen Bildbereichen konnte keine gemeinsame Bildinformation durch die Transformation zugeordnet werden.

[2] Ergebnis der Studienarbeit von Peter Decker

2.2 Feature Extraktoren

Feature Extraktoren haben das Ziel eindeutige Merkmale in Bildern zu finden. Dabei ist es nicht von Bedeutung, ob sie für den Betrachter als sinnvoll erscheinen oder nicht, sofern sie sich gut und eindeutig beschreiben lassen. Sie sollten weiterhin die Eigenschaft besitzen, dass sie auch unter verschiedenen Aufnahmebedingungen zum selben Ergebnis führen. Rotationsinvarianz, Translationsinvarianz und Beleuchtungsinvarianz gelten deswegen als erstrebenswerte Vorgaben. In den folgenden Abschnitten werden nun bekannte Merkmalsextraktionsoperatoren vorgestellt.

2.2.1 Moravec-Operator

Der Moravec-Operator [Mor80] bestimmt ein Feature über die Differenzen zwischen den Pixelintensitäten in einer lokalen Nachbarschaft w_{ij}.

$$w_{ij} \quad = \quad [w_{ij}, \mu\nu]_{\mu,\nu=0,\cdots 6} \tag{2.10}$$

Dabei werden jeweils in horizontaler (m_{xxij}), vertikaler (m_{yyij}), diagonaler (m_{xyij}) und gegen-diagonaler Richtung (m_{yxij}) die Summe der 6 Teildifferenzen über alle Zeilen und Spalten in einem Fenster w_{ij} bestimmt. Als Resultat für die Interessenkarte H_{ij}, die die Koordinaten für interessante Features in einem Bild speichert, nimmt man das minimale Ergebnis dieser vier Summen. Das Minimum im lokalen Fenster für Gleichung 2.15 wird

als Merkmal an der Stelle (i, j) akzeptiert, sofern es über einer gegebenen Schwelle liegt.

$$m_{xxij} = \sum_{\mu=0}^{5}\sum_{\nu=0}^{5}(w_{ij,\mu\nu} - w_{ij,\mu\nu+1})^2 \tag{2.11}$$

$$m_{yyij} = \sum_{\mu=0}^{5}\sum_{\nu=0}^{5}(w_{ij,\mu\nu} - w_{ij,\mu+1\nu})^2 \tag{2.12}$$

$$m_{xyij} = \sum_{\mu=0}^{5}\sum_{\nu=0}^{5}(w_{ij,\mu\nu} - w_{ij,\mu+1\nu+1})^2 \tag{2.13}$$

$$m_{yxij} = \sum_{\mu=0}^{5}\sum_{\nu=0}^{5}(w_{ij,\mu+1\nu} - w_{ij,\mu\nu+1})^2 \tag{2.14}$$

$$H_{ij} = min\{m_{xxij}, m_{yyij}, m_{xyij}, m_{yxij}\} \tag{2.15}$$

2.2.2 Kanade - Lucas - Tomasi Operator (KLT)

Der KLT - Operator [ST94] hat als Grundlage zur Erstellung der Interessenkarte folgende Vorverarbeitungsschritte. Zu Beginn wird das Bild f jeweils in horizontaler d_x und vertikaler Richtung d_y abgeleitet. Die beiden abgeleiteten Bilder f_x, f_y werden jedes für sich mit einem Gaußfilter G_σ mit einer Standardabweichung von $\sigma = 0.7$ geglättet. Die beiden geglätteten Bilder $\tilde{f}_x, \tilde{f}_y$ werden in einer Struktur-Matrix M_{ij}, Gleichung 2.18, eingetragen.

$$\tilde{f}_x = f_x \star G_\sigma = f \star d_x \star G_\sigma \tag{2.16}$$

$$\tilde{f}_y = f_y \star G_\sigma = f \star d_y \star G_\sigma \tag{2.17}$$

$$\boldsymbol{M_{ij}} = \begin{pmatrix} M_{ij,11} & M_{ij,12} \\ M_{ij,21} & M_{ij,22} \end{pmatrix} \tag{2.18}$$

mit

$$M_{ij,11} = \sum_{\mu\nu\epsilon N_{ij}} \tilde{f}^2_{x,\mu\nu} \quad (2.19)$$

$$M_{ij,12} = \sum_{\mu\nu\epsilon N_{ij}} \tilde{f}_{x,\mu\nu} * \tilde{f}_{y,\mu\nu} \quad (2.20)$$

$$M_{ij,21} = \sum_{\mu\nu\epsilon N_{ij}} \tilde{f}_{x,\mu\nu} * \tilde{f}_{y,\mu\nu} \quad (2.21)$$

$$M_{ij,22} = \sum_{\mu\nu\epsilon N_{ij}} \tilde{f}^2_{y,\mu\nu} \quad (2.22)$$

Der KLT-Operator wertet nun die Eigenwerte der Matrix M_{ij} aus, wobei die Eigenwerte λ_1 und λ_2 Aussagen über die jeweilige lokale Struktur des Pattern ermöglichen und somit ein Kriterium dafür liefern, ob ein interessantes Feature vorliegt. Folgende Aussagen können an Hand der Eigenwerte in Tabelle 2.2 getroffen werden.

$$\lambda_{ij,1/2} = \frac{1}{2}(M_{ij,11} + M_{ij,22} \pm \sqrt{(M_{ij,11} - M_{ij,22})^2 + 4M^2_{ij,12}}) \quad (2.23)$$

$$min(\lambda_{ij,1}, \lambda_{ij,2}) > Schwellwert \quad (2.24)$$

Falls beide Eigenwerte kleiner als die vorgegebene Schwelle sind, liegt eine homogene Region vor, in welcher keine Gradientenübergänge vorhanden sind. Ist λ_1 größer als der gewählte Schwellwert, so handelt es sich um eine Kante, da ein starker Gradientenübergang in einer Richtung vorliegt. Erfüllen beide Werte die Bedingung so handelt es sich um ein interessantes Feature, das sowohl in horizontaler als auch in vertikaler Richtung hohe Gradientenübergänge aufweist. In Abbildung 2.6 ist der letzte Fall jeweils für ein ideales Merkmal und reales Merkmal eines Bildes dargestellt.

2.2.3 Harris Corner - Operator

Der Harris Corner - Operator [HS88] wendet dieselben Vorverarbeitungsschritte an wie der KLT - Operator aus Unterabschnitt 2.2.2. Die Auswertung der Matrix $M_{i,j}$ in Gleichung 2.18 erfolgt jedoch durch Berechnung der Determinante (det) und Spur ($trace$) in Gleichung 2.25. Auch hier werden die Werte in eine Interessenkarte geschrieben und als Feature akzeptiert, sofern sie eine vorgegebene Schwelle überschreiten. Der Parameter κ kann dabei i.

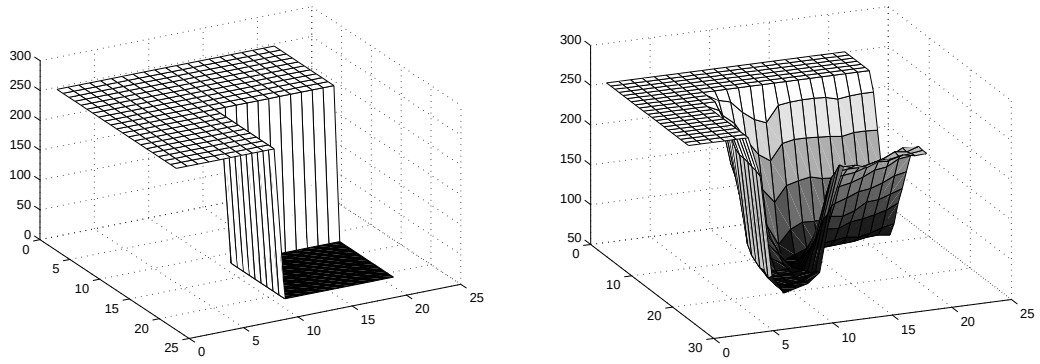


Bild 2.6: [v.l.n.r]Gegenüberstellung eines idealen Features mit einem fixen weiß-schwarz Übergang und eines Features aus einer realen Aufnahme

	λ_1	λ_2	lokale Struktur
$>$ Schwellwert	nein	nein	homogen
$>$ Schwellwert	ja	nein	Kante
$>$ Schwellwert	ja	ja	Ecke / Feature

Tabelle 2.2: In Abhängigkeit von einem Schwellwert entscheidet der KLT-Operator, ob es sich um ein interessantes Feature handelt

A. Werte zwischen 0.02 und 0.25 annehmen. In der Praxis hat sich jedoch ein Wert von 0.04 durchgesetzt.

$$\boldsymbol{H_{ij}} \quad = \quad det(\boldsymbol{M_{ij}}) - \kappa * trace(\boldsymbol{M_{ij}^2}) \tag{2.25}$$

2.2.4 SIFT - Operator

Der SIFT - Operator (scale-invariant-feature-transformation) nach David G. Lowe [Low04] verfolgt im Vergleich zu den vorhergehenden Feature-Operatoren, insofern einen anderen Ansatz, dass er jedem gefundenem Feature direkt beschreibende Eigenschaften hinzufügt, mittels welcher sie identifiziert werden können. Sie überzeugen vor allem durch ihre Rotations- und Skalierungsinvarianz und durch ihre teilweise Invarianz bzgl. Beleuch-

tung und Kamerastandpunkt. Der generelle Ablauf des Algorithmus erfolgt dabei über vier Schritte.

1. **Die Extremumdetektion im Skalenraum** wird mit Hilfe einer Gaußdifferenzfunktion bestimmt. Man erhält mögliche Kandidaten, die invariante Eigenschaften besitzen. Diese Vorabfilterung reduziert, die Menge an Schlüsselpunkten, die durch weitere Schritte genauer transformiert werden und einen eindeutigen Deskriptor erhalten. Dieser erste Schritt wird auf allen Bilddaten und Skalierungsstufen angewandt. Die Bestimmung eines lokalen Extremums $(\mathrm{argmin}, \mathrm{argmax})$ erfolgt durch einen Vergleich, des aktuellen Pixel mit seinen Nachbarschaften in der aktuellen, der darüber und der darunter liegenden Skalierungsstufe. Es wird akzeptiert, sofern es kleiner bzw. größer als alle 26 Nachbarpixel ist. Abbildung 2.7 zeigt die ersten beiden Stufen des Skalenraums und die resultierenden Differenzbilder von benachbarten Gaußbildern, sowie die Bestimmung eines lokalen Extremums

2. **Die Exakte Schlüsselpunkt-Lokalisierung** für jeden der gefilterten Punkte bzgl. erfolgt mit Hilfe eines Orts- und Skalenmodells. Die ausgewählten Punkte müssen eine gewisse Stabilität aufweisen, so dass z. B. Punkte an Kanten oder mit niedrigem Kontrast zurückgewiesen werden, da diese beim Wiederfinden in anderen Datensätze durch Ambiguitäten bzw. durch Rauschen beeinflusst werden. Als Basis dient die Hesse-Matrix aus Gleichung 2.18, um zu entscheiden, ob ein bis hier selektiertes Merkmal weiterhin als relevant betrachtet wird.

3. **Das Zuweisen von Orientierungen pro Schlüsselpunkt** geschieht durch den zum lokalen Punkt gehörenden Gradienten. Die Orientierung wird über ein Histogramm mit 36 Behältern ermittelt. Jeder Histogrammeintrag wird mit seiner Gradientenstärke und einer Gaußfunktion gewichtet. Maximalwerte entsprechen prägnanten Richtungen und werden zusätzlich detektiert, sofern sie nicht mehr als 20% vom Maximum abweichen. Daraus folgt, dass mehrere Schlüsselpunkte an derselben Stelle möglich sind. Sie unterscheiden sich nur in ihrer Orientierung, repräsentieren dafür aber ein sehr auffälliges Merkmal. Alle weiteren Operationen für jedes Merkmal, auf transformierten Bilddaten, werden relativ zur Zuweisung von Orientierung, Skalierung und Ort durchgeführt. Damit wird letztendlich die Invarianz der einzel-

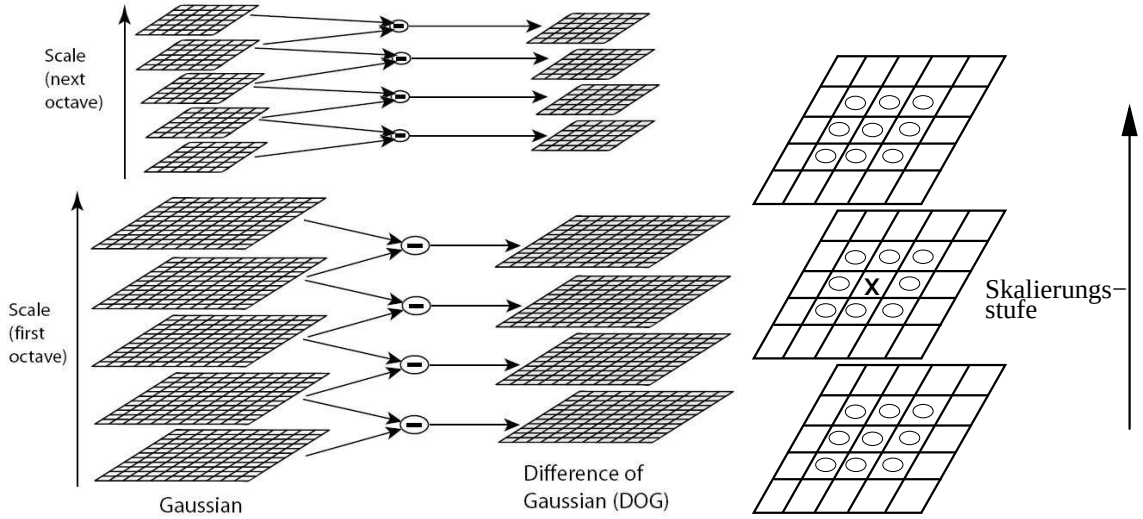


Bild 2.7: Rechts zeigt die Stufen des Skalenraums und die sich daraus ergebende Berechnung der jeweiligen Gaußdifferenz aus benachbarten Gaußbildern. Die Gaußbilder werden dabei mehrfach mit sich selbst gefaltet. Dies wird wiederholt sich für die nächste Stufe im Skalenraum. Die Bilder werden mit einem Faktor 2 kleinskaliert. Links wird die Bestimmung eines lokalen Extremums für ein Pixel und seine 26 Nachbarn dargestellt.

nen Transformationsschritte erreicht.

4. **Die Generierung der Schlüsselpunktbeschreibung** wird durch die Bestimmung der lokalen Gradientenstärken und Gradientenorientierungen für die jeweilige Skalierungsstufe in der Nähe eines Punktes erreicht. Diese werden mit einer Gaußfunktion gewichtet und in einem Orientierungshistogramm akkumuliert. Das Histogramm fasst, um den Ort des Merkmals, jeweils die Werte aus 8×8 Unterregionen zusammen. Die Größe eines Eintrags entspricht der Summe der Gradientenstärke in der Nähe dieser Richtung. Die Beschreibung des Schlüselpunktes wird in einem 128 großen Vektor gespeichert. Um außerdem den Einfluss von Beleuchtungsänderungen zu vermeiden, wird der Vektor zusätzlich normiert. Der endgültige Vektor enthält also eine Beschreibung, die invariant gegenüber Skalierung, Rotation, Beleuchtung und verschiedenen Blickwinkeln ist.

In Abbildung 2.8 kann man die Art und Weise beobachten, welche Features für den jeweiligen Operator als interessant gelten. Dabei wurden die jeweils 200 stärksten Features eines Operators ausgewählt. Auffallend dabei ist vor allem, dass der Moravec-Operator häufiger an Kanten anschlägt im Vergleich zum KLT-Operator und Harris-Corner. KLT und Harris hingegen liefern auch intuitiv für den Betrachter vernünftige Merkmalsmengen. Die SIFT-Schlüsselpunkte vermitteln eher den Eindruck, dass sie oft etwas weiter entfernt vom eigentlichen wahrgenommen Merkmal in der Szene liegen. Das kommt jedoch durch die mehrfache Skalierung im Raum zu Stande, so dass das Merkmal vom eigentlichen Punkt *wegläuft*, sich aber weiterhin über seinen eindeutigen Deskriptor sehr genau beschreiben und zuordnen lässt.

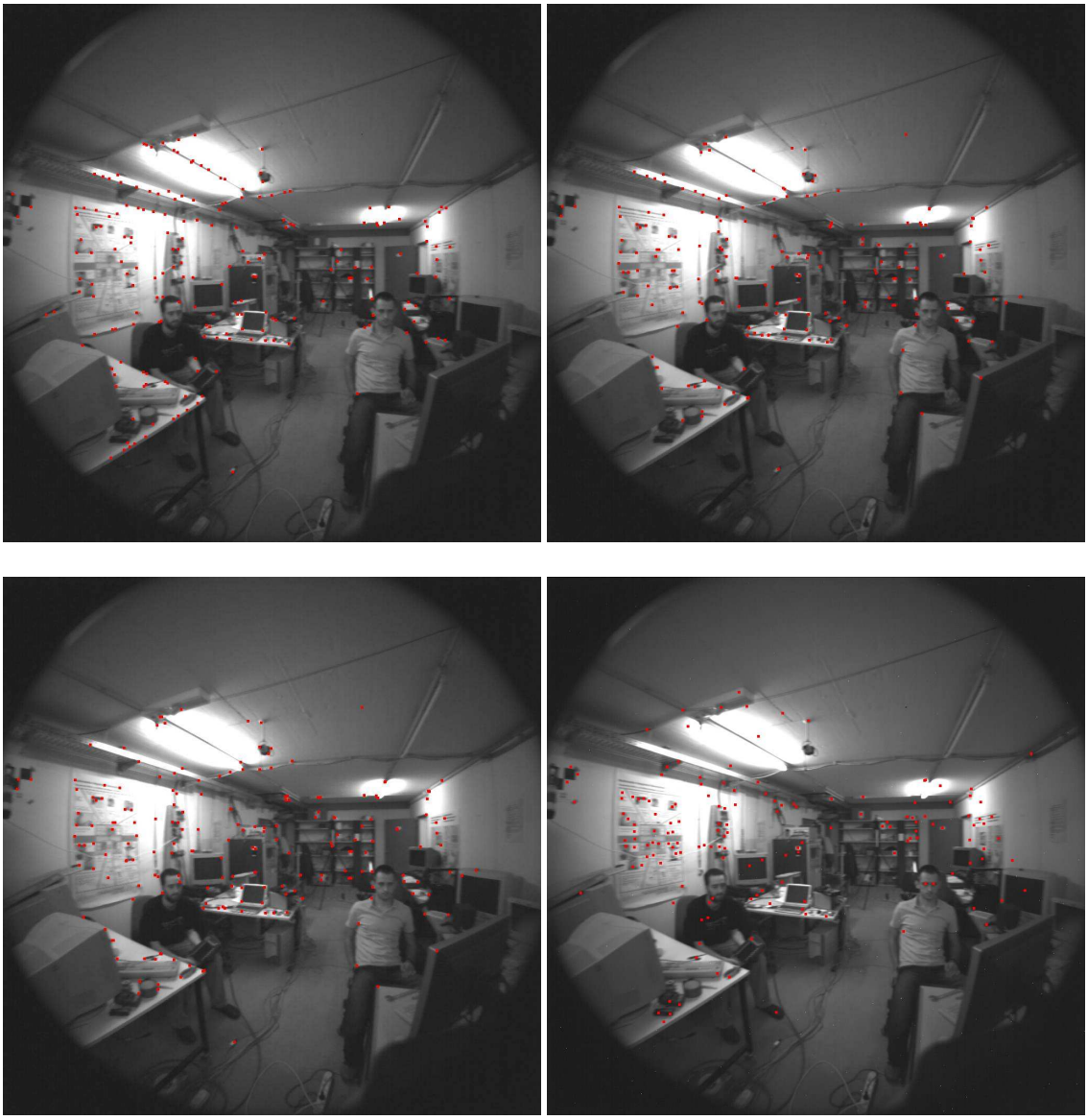


Bild 2.8: [v.l.n.r.,v.o.n.u.] Gegenüberstellung der jeweiligen Extraktoren mit den 200 stärksten Features. MORAVEC, KLT, HARRIS, SIFT

2.3 Matching Metriken

Die Korrespondenzbestimmung wird oft unter verschiedenen Voraussetzungen durchgeführt. Je nach Aufgabenstellung verlangt das Ergebnis nach einer großen Anzahl von bekannten Punktkorrespondenzen bzw. nach wenig bekannten Punktkorrespondenzen. Mittels der Operatoren aus Abschnitt 2.2 lassen sich z. B. Punkte gezielt für die Matchingaufgabe auswählen.

Als Ergebnis erhält man generell eine dichte bzw. eine dünne Tiefenkarte (engl.: dense / sparse depth map), wobei die Aussagekraft der dichten Karte bzgl. der Raumgeometrie größer ist. Die dünne Tiefenkarte ermöglicht die Bestimmung der Bewegung eines Kamerasystems von gezielt ausgesuchten 3D-Punkten, man versucht dabei aus der Bewegung die Struktur der Szene zu erkennen. Die beiden Strategien die zur Erkennung der Raumgeometrie führen sollen, bezeichnet man als globales und lokales Matching. Die globalen Matching Verfahren versuchen jedem Bildpunkt (i, j) im linken Kamerabild einem Bildpunkt (i, j) im rechten Kamerabild zu zuordnen und somit für jeden Pixel einen 3D - Punkt zu errechnen. Daraus lässt sich leicht eine dichte Tiefenkarte bestimmen.

Bekannte Verfahren sind Dynamic Programming [OK85], [BT98], GraphCuts [BVZ01], [KZ01] und Belief Propagation [SZS03]. Als Beispiel einer Tiefenkarte sei hier auf Abbildung 2.9 der Universität Tsukuba mit einem vermessenem Grundtiefenbild (engl.: ground truth) verwiesen. Eine Vielzahl an Evaluationsergebnissen finden sich vor allem auf der Internetpräsenz von Scharstein und Szeliski [SSZ01][3]. In dieser Arbeit werden jedoch die lokalen Matching Metriken vorgestellt, die sich nur auf einzelne Bildausschnitte – Fenster, Pattern, Blöcke, Regionen – beziehen. Diese ermöglichen eine schnellere Berechnung der gewünschten Korrespondenz, für ein kleine Anzahl an ausgewählten Features, die das Zentrum des jeweiligen Blocks bilden, der zum Matching herangezogen wird.

Aus der Aufgabenstellung erschließt sich hieraus die Wahl für den weiteren Verlauf der Arbeit zu Gunsten der lokalen Matching-Metriken wie diese auch von Nister [NBN06] und Hirschmüller [Hir03] angewandt wird. Da sich die Raumgeometrie somit aus der Bewegung des Stereosystems entsprechend bestimmen lässt. Ein Vergleich zu den beiden Herangehensweise von Nister und Hirschmüller erfolgt in Abschnitt 3.6.

[3] `http://www.middlebury.edu/stereo/`

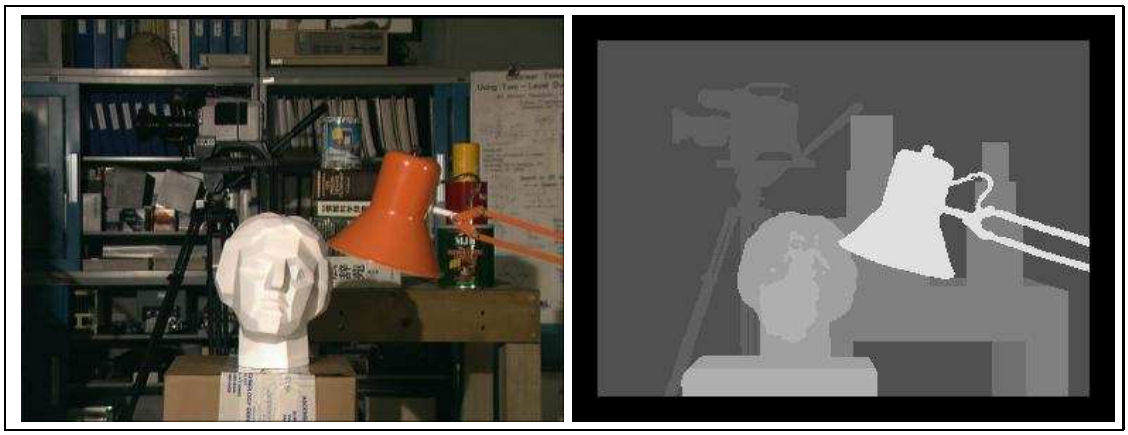


Bild 2.9: [v.l.n.r] linkes Kamerabild, Grundtiefenbild der Universität Tsukuba: helle Bereiche geben eine größere Disparität an, d. h. die Objekte befinden sich näher an der Kamera als dunkle Bereiche der Tiefenkarte.

Nachfolgend werden in Unterabschnitt 2.3.1 mehrere lokale Matchingalgorithmen beschrieben und Unterabschnitt 2.3.2 gibt einen Überblick an aktuellen globalen Matchingverfahren.

2.3.1 Lokale Korrespondenzanalyse

Normalized Cross Correlation - NCC Die Gleichung 2.26 gibt die Berechnung für die Bestimmung der Korrelation an: Mit i, j als Koordinatenindex, $\bar{I}_{1,2}$ als Mittelwert des jeweiligen Pattern, (d, e) repräsentieren die Shiftparameter für den Suchraum im zweiten Bild, in welchem die beste Korrespondenz für ein Pattern bestimmt wird. Die normalisierte Kreuzkorrelation ist robust gegenüber additivem und multiplikativem Rauschen. Bei perfektem Matching zwischen zwei Pattern liefert sie als Ergebnis 1. Die Bestimmung der Korrespondenz ist jedoch teuer. In Abschnitt 4.3 erfolgt eine genaue Angabe zu den Rechenkosten.

$$\frac{\sum\limits_{i,j}(I_1(i,j) - \bar{I}_1 * (I_2(i+d, j+e) - \bar{I}_2)}{\sqrt{\sum\limits_{i,j}(I_1(i,j) - \bar{I}_1)^2 * \sum\limits_{i,j}(I_2(i+d, j+e) - \bar{I}_2)}} \tag{2.26}$$

Sum of absolute difference - SAD Die Summe der absoluten Differenz zwischen zwei Pattern erlaubt eine schnelle und effiziente Berechnung um diese auf eine mögliche Korrespondenz hin zu überprüfen. Bei idealer Korrespondenz liefert sie als Ergebnis 0.

$$\sum_{i,j} |I_1(i,j) - I_2(i+d, j+e)| \tag{2.27}$$

Sum of squared difference - SSD Die Summe der Quadratdifferenzen zur Bestimmung der Korrespondenz bestraft mögliche Fehlmatches stärker als Gleichung 2.27, da die Abweichung quadratisch eingeht. Die Berechnung zwischen zwei Pattern wird in Gleichung 2.28 angegeben.

$$\sum_{i,j} (I_1(i,j) - I_2(i+d, j+e))^2 \tag{2.28}$$

Normalized sum of squared difference - NSSD Die Normalisierung der SSD führt zur einer nochmaligen Steigerung der Genauigkeit und macht sie vergleichbar u. a. zu Gleichung 2.26. Gleichung 2.29 beschreibt die normalisierte Formel zur Berechnung. Bei ähnlichen Pattern konvergiert das Ergebnis gegen 0. Wie auch die vorangegangenen Metriken ist auch die NSSD robust gegenüber additivem und multiplikativem Rauschen.

$$\sum_{i,j} \left(\frac{(I_1(i,j) - \bar{I_1})}{\sqrt{\sum_{i,j} (I_1(i,j) - \bar{I_1})^2}} - \frac{(I_2(i+d, j+e) - \bar{I_2})}{\sqrt{\sum_{i,j} (I_2(i+d, j+e) - \bar{I_2})^2}} \right)^2 \tag{2.29}$$

Rank - Transformation Die Rank-Transformation ist ein non-parametrisches Modell, dass in [ZW00] beschrieben wird. Dabei wird ein lokales Pattern beleuchtungsinvariant und rotationsinvariant mit Gleichung 2.30 transformiert. Das Pixel wird durch eine $<$ - Relation zu seiner lokalen Nachbarschaft beschrieben. Der transformierte Wert ergibt sich aus der Anzahl an Nachbarpixeln, die kleiner als der betrachtete Pixel sind. Eine vermutete Korrespondenz zwischen zwei Punkten ergibt aus der Differenz der transformierten Pixelwerte, im Idealfall gleich 0. Da jedoch der Informationsverlust sehr hoch ist, eignet es sich nicht für das später hier angewandte Tracking-Verfahren, wie in Abschnitt 4.3

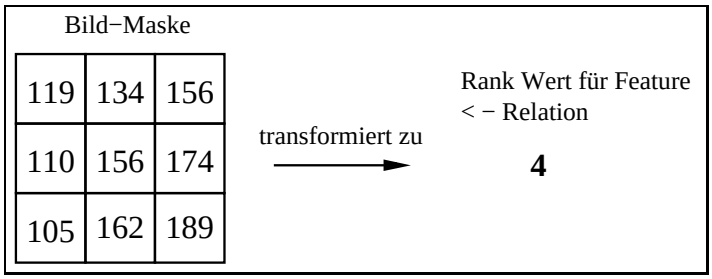


Bild 2.10: Beispiel: 3×3 Pattern mit Rank transformiert.

gezeigt wird. Lediglich bei vorher erfolgter Rektifizierung der Bildpaare und unter Ausnutzung der epipolaren Bedingung kann man mit brauchbaren Ergebnissen für das Stereo-Matching rechnen, da der extrem kleine Suchraum mögliche Mehrdeutigkeiten zwischen einzelnen Pixeln reduziert.

$$\sum_{i,j}(I_1'(i,j) - I_2'(i+d, j+e))$$
$$I'(i,j) = \sum_{m,n}(I_k(m,n) < I_k(i,j)) \tag{2.30}$$

Abbildung 2.11 zeigt ein rank-transformiertes Bild. Auffällig ist dabei, dass die ursprüngliche Bildstruktur komplett erhalten bleibt. Der große Vorteil dieser Transformation liegt sicherlich in der sehr schnellen und einfachen Berechnung der Patternbeschreibung und den o. g. Invarianzen bezüglich Beleuchtung und Rotation.

Census - Transformation Diese Transformation führt die Idee aus Gleichung 2.30 weiter und erhält dabei zusätzlich die lokale Struktur des Blocks. Es erfolgt eine Bit-Kodierung der Nachbarschaft in Relation zum Zentrumspixel, welcher mittels eines Feature-Operators bestimmt wird. Pixel, die kleiner als das Zentrumspixel sind, werden mit einer 1 kodiert und Pixel, welche größer sind, werden entsprechend mit einer 0 kodiert. Die inverse Kodierung des resultierenden Bit-Musters ist ebenfalls möglich, ändert aber nichts am Informationsgehalt der Transformation. Beim Vergleich auf Korrespondenz zwischen zwei Blöcken wird die Hamming-Distanz berechnet. Das Ergebnis der Hamming-Distanz gibt die Anzahl der unterschiedlichen Bits zwischen den beiden kodierten Pattern an. Im Ideal-

Bild 2.11: Ranktransformiertes Bild mit einer Maskengröße von 7×7. Der Intensitätsbereich ist nach der Transformation $[0, 48]$.

fall liefert die Hamming-Distanz als Ergebnis 0. Die genaue Art und Weise der Transformation und der Korrespondenzbestimmung gibt nochmals Gleichung 2.31 an.

$$\begin{aligned} &\sum_{i,j} HAMMING(I_1'(i,j), I_2'(i+d, j+e)) \\ I'(i,j) &= BITSTRING_{m,n}(I_k(m,n) < I_k(i,j)) \end{aligned} \tag{2.31}$$

Das Beispiel in Abbildung 2.12 zeigt eine Transformation einer 3×3 Maske in das Bit-Pattern, das zum Vergleich herangezogen wird. Im Vergleich zur Rank-Transformation vervielfacht sich hier der Informationsgehalt des zu beschreibenden Pattern in Abhängigkeit zur Größe des Korrelationsfensters, so dass aber auch die Beschreibung des Blocks eindeutiger wird und somit Mehrdeutigkeiten vermieden werden. Das 3×3 Pattern (2.32) macht nochmals den Vorteil des non-parametrischen Ansatzes gegenüber parametrischen Modellen deutlich.

$$\begin{matrix} 119 & 134 & 156 \\ 110 & 156 & 174 \\ 105 & 162 & \alpha \end{matrix} \tag{2.32}$$

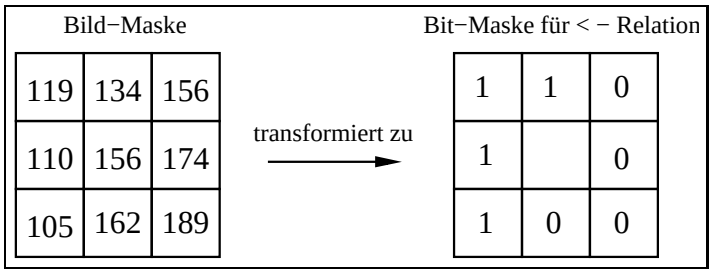


Bild 2.12: Beispiel: 3×3 Pattern mit Census transformiert.

mit $\alpha \in [0, 255]$. Der Mittelwert dieses Pattern schwankt zwischen 124 und 152, die Varianz liegt zwischen Extremen von 624 und 2756. Wo hingegen der Rank-Wert sich zwischen 4 und 5 bewegt und das Census Bit-Pattern kippt entsprechend an einer einzelnen Stelle seiner Kodierung $\{1, 1, 0, 1, 0, 1, 0, \alpha\}$, je nachdem ob die Pixelintensität α kleiner oder größer als das Zentrumspixel ist. Dies verdeutlicht den Vorteil der beiden Transformationen gegenüber parametrischen Modellen, die einer viel stärkeren Abweichung unterliegen bei additivem und multiplikativem Rauschen. Modifizierte Varianten der Census-Transformation von Ojala und Pietikäinen [OPX00], [OPM02] versuchen auch die in der Rank-Transformation enthaltene Rotationinvarianz wiederherzustellen und somit das Matching zu verbessern. Bei Ojala und Pietikäinen werden sie als Monotonieoperator für Rank und als Local Binary Pattern für Census bezeichnet. Leider konnten die dort aufgeführten Verbesserungen des Matching von Balthasar [Bal06] nicht bestätigt werden.

Least-Square Matching Das Matching der kleinsten Quadrate bestimmt Punktkorrespondenzen mit Subpixelgenauigkeit. So lassen sich bereits akzeptierte Punkte aus einen der vorhergegangen Methoden nochmals genauer bestimmen. Um die exakten Koordinaten zwischen zwei Korrespondenzen bestimmen zu können, wird das überbestimmte Gleichungssystem

$$\boldsymbol{A} * \boldsymbol{x} \;=\; \boldsymbol{l} \tag{2.33}$$

mit einem iterativem Ansatz gelöst. Das mögliche Pattern wird dabei solange affin transformiert bis der bestmöglichste Fit nach einer vorgegebenen Anzahl an Iterationen erreicht

wird. Der Lösungsvektor $\boldsymbol{x}$ wird dabei mit einem Least-Square Schätzer wie folgt berechnet.

$$\boldsymbol{x} \;=\; (\boldsymbol{A}^T * \boldsymbol{A})^{-1} * \boldsymbol{A}^T * \boldsymbol{l} \tag{2.34}$$

Genaue Ergebnisse zu Initialisierungswerten, Abbruchkriterien und Genauigkeit können bei A.Gruen [Gru85] nachgelesen werden. Im Vergleich zu allen anderen Matching Metriken ist der Ansatz der kleinsten Quadrate sicherlich der Genaueste, aber auch der Zeitintensivste.

2.3.2 Globale Korrespondenzanalyse

Dynamic Programming Die Idee der dynamischen Programmierung besteht darin, ein großes komplexes Problem in viele kleine Probleme zu zerlegen. Die vereinfachten Probleme wiederum sind mit geringerem Rechenaufwand und verminderter Komplexität besser zu lösen. Bezogen auf die Bestimmung von Punktkorrespondenzen in Stereobildern bestimmen die kleinen lokalen Berechnungen über mehrere Stufen eine globale Kostenfunktion. Die lokalen Kosten werden dabei mit einer der Korrespondenzmethoden aus Unterabschnitt 2.3.1 bestimmt. Die Punktkorrespondenz wird also für jedes Pixel bestimmt um somit eine dichte Tiefenkarte für das Stereobildpaar zu erhalten. Die Zerlegung in kleinere Probleme entspricht folglich einer Punkt-Scanline-Bild Reihenfolge bezüglich der Korrespondenzkostenhierarchie.

Der Vorteil dieses globalen Ansatzes beim Stereo-Matching ist, dass man stützende, globale Bildinformationen beim Betrachten von lokalen Regionen hat und dadurch die Stabilität des Matching steigern kann. Probleme entstehen jedoch, wenn kleine lokale Fehler in der Scanline weiter propagiert werden und somit korrekte Zuordnungen beeinflussen und das Scanline-Matching verfälschen. Dies beschreibt auch schon eine weitere Problemstelle. Die dynamische Programmierung ist nur in der Lage horizontale Bedingungen zu berücksichtigen und lässt die vertikale Konsistenz der Pixelkorrespondenzen außen vor.

In [OK85] und [CHRM96] werden zwei gängige Ansätze für die dynamische Programmierung beschrieben. Für jede Scanline wird ein Disparity - Space - Image ausgewertet. Abbildung 2.13 zeigt ein solches Disparitätsweitenbild. Zwischen linkem und rechtem

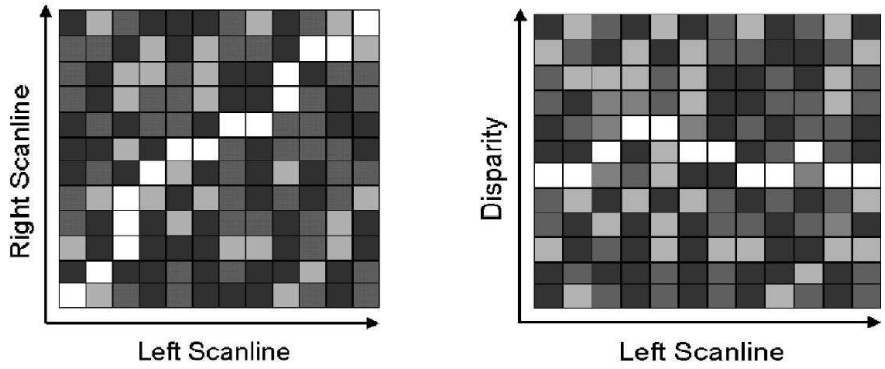


Bild 2.13: [v.l.n.r.]Darstellung eines Disparity-Space-Image mit linker und rechter Scanline nach[OK85] und linker Scanline und Disparitätsweite nach [CHRM96]. Beispiel aus [BBH03].

Bild wird für jede Zeile durch Suchen eines optimalen Pfades von der linken unteren Ecke, diagonal verlaufend, hin zur rechten oberen Ecke des Graphs, die Korrespondenz zwischen den beiden Zeilen bestimmt.

Ein ähnlicher Ansatz berechnet die Tiefenkarte aus dem optimalen Weg zwischen linker Bildzeile und der maximalen Disparität (engl.: disparity range). Der optimale Pfad wird nun von links nach rechts gehend gesucht. Dies wird für jede Bildzeile umgesetzt, so dass sich die komplette Tiefeninformation für das gesamte Bild am Ende berechnen lässt.

Intrinsische Kurven Ein andere Idee verfolgt Tomasi [TM98] zur Bestimmung der Korrespondenz. Er orientiert sich dabei an so genannten intrinsischen Kurven der jeweiligen Scanline. Dabei legt er die abgeleiteten Intensitätskurven der entsprechenden Zeilen übereinander. Die überlagerten Kurven werden gegeneinander geplottet. Im Idealfall ergibt sich eine deckungsgleiche Linie. Aufgrund von Rauschen muss jedoch eine Suche nach der Korrespondenz im Nearest-Neighbour-Bereich stattfinden. Ein einfaches Beispiel zeigt Abbildung 2.14

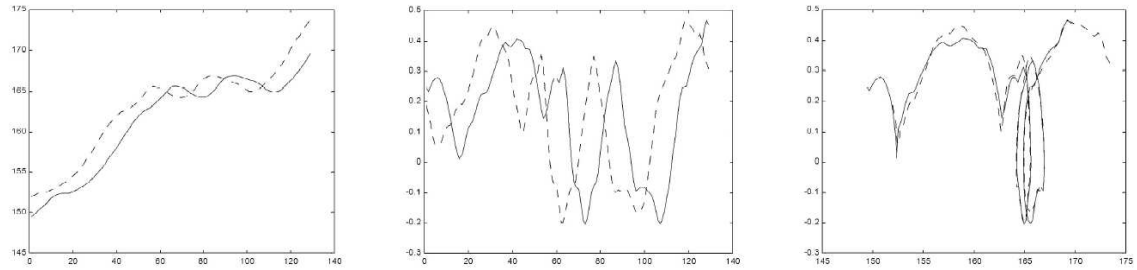


Bild 2.14: [v.l.n.r.] Linke und rechte Scanline Intensitäten, deren Ableitungen und die dazugehörigen intrinsischen Kurven, jeweils gegeneinander geplottet. Beispiel aus [BBH03]

Graph Cuts Anders als die dynamische Programmierung versuchen Roy und Cox mit Graph - Repräsentationen [RC98] auch die vertikale Konsistenz beim Zuordnen von Korrespondenzen zu berücksichtigen. Dafür wird ein gerichteter Graph $G = (V, E)$ aufgebaut, wobei V der Knoten ist und E die Kante zwischen zwei Knotenpunkten. Knoten repräsentieren dabei die Pixel. Die Art und Weise wie der Graph für das Disparity-Space-Image aufgebaut wird, kann nach den gleichen Kriterien wie in Abbildung 2.13 erfolgen. Die Definition eines Knoten lautet,

$$V = V^* \cup \{s, t\} \tag{2.35}$$

$$V^* = \{(x, y, d), x \in [0, x_{max}], y \in [0, y_{max}], d \in [0, d_{max}]\} \tag{2.36}$$

wobei s die Quelle (engl.: source) und t die Senke (engl.: sink) ist. Aus der Definition folgt, das die jeweiligen Achsen des Graphen die Bildhorizontalen, die Bildvertikalen und die Disparitätsweite enthalten. Eine Kante wird wie folgt definiert:

$$E = \left\{ \begin{array}{rcl} (u, v) \in V^* \chi V^* & : & \|u - v\| = 1 \\ (s, (x, y, 0)) & : & x \in [0, x_{max}] \\ ((x, y, d_{max}, t) & : & y \in [0, y_{max}] \end{array} \right\} \tag{2.37}$$

Intern führt dies zu einem sechsfach verbundenen Netz, die Vektornorm aus Gleichung 2.37 garantiert außerdem, dass Knotenpaare verbunden bleiben[4]. Die Kosten von der Quelle zur Senke berechnen sich mit Hilfe von lokalen Matching-Metriken aus Unterabschnitt 2.3.1,

[4]Darius Burschka et al.[BBH03]

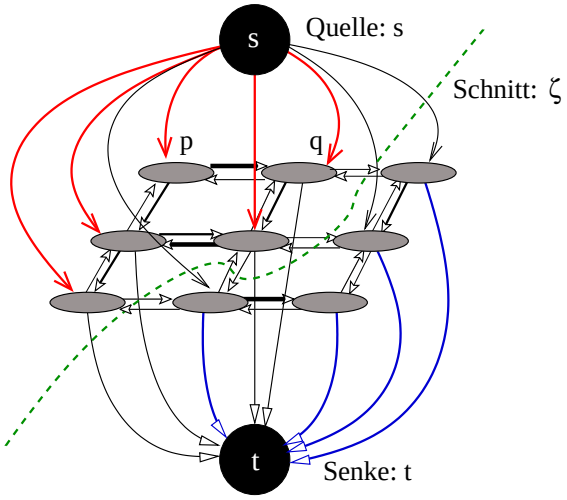


Bild 2.15: Beispiel nach [KZ01] für einen gerichteten Graphen, der einen Schnitt für die minimalen Kosten enthält. Die Kantendicke zwischen Knoten steht für die Kosten, die anfallen, um von einem Knoten zum Nachbarknoten zu gelangen.

wobei sich die Teilkosten zwischen benachbarten Knoten $< p, q >$ und $< q, p >$ unterscheiden können. Die Zuordnung von Punkten erfolgt nun mit Hilfe eine minimalen Schnittes ζ. Ein Schnitt ζ (engl.: cut) durch den Graphen entspricht einer Teilung eines Knoten V in zwei Untermengen, so dass $s \in V^s$ und $t \in V^t$ gilt. In Abbildung 2.15 ist ein Beispiel für den Schnitt durch einen gerichteten Graphen dargestellt. Die Kosten für ζ ergeben sich aus den gewichteten Summen der Kanten, die einen Knoten in V^s mit einem Knoten in V^t verbindet. Der Schnitt mit den minimalen Kosten muss bestimmt werden, um die optimale Korrespondenz zu finden. Die Bestimmung des maximalen Flusses zwischen den Terminals liefert den Schnitt mit den geringsten Kosten bzw. der minimale Schnitt liefert den maximalen Fluss von der Quelle (s) zur Senke (t). Dies ist analog zur Berechnung der Scanline mit Hilfe der dynamischen Programmierung, allerdings erweitert auf drei Dimensionen, wie in Definition Gleichung 2.37 festgelegt. Die Erweiterung auf drei Dimensionen hat zur Folge, dass das Matching nun auch die vertikale Konsistenz berücksichtigt. Die daraus resultierende Tiefenkarte erweist sich als wesentlich genauer.

Eine neuerer Ansatz von Boykov [BVZ01] greift die Graphenrepräsentation auf und op-

timiert diese durch eine erweiterte Darstellung und Bestimmung des optimalem Pfades. Dies führt zu Laufzeitverbesserungen von Faktor 2 bis hin zu Faktor 5 auf den dort angegebenen Testbildern (z. B. Abbildung 2.9) im Vergleich zum Standardansatz.

Kapitel 3

Intra- und Inter-Matching von Punktmengen

Das größte Problem, das sich aus der Aufgabenstellung ergibt, ist das nicht Vorhandensein von kalibrierten und rektifizierten Bilddaten. Der Aufbau erlaubt es daher nicht, den Suchraum beim Matching auf eine Linie einzuschränken. Die daraus resultierenden großen Suchräume verlangsamen den Matchingprozess entscheidend.

Bei idealisierten Aufnahmevoraussetzungen hingegen kann man sich mehrere Gegebenheiten zur Einschränkung des Suchraumes zu Nutze machen. Neben der Reduzierung der Suche auf ein eindimensionales Problem durch die epipolare Bedingung, kann man sich weiterhin zu Nutze machen, dass Punkte im linken Bild mit einer hohen x-Koordinate sich im rechten Bild auf jeden Fall links von dieser Bildspalte befinden müssen. D. h. , das sie eine kleinere x-Koordinate aufweisen, da sonst die Disparitätsbedingung verletzt wird, die durch die gemeinsame Bildebene aus dem idealen Modell vorgegeben wird.

Von Vorteil für die Suche ist es außerdem, wenn die Möglichkeit besteht Informationen der gemachten Bewegung, eines weiteren Sensors oder einer vorher definierten Bewegung miteinfliessen zu lassen und somit bereits einen vorgegebenen Startpunkt für die Suche zu erhalten. Ein weiterer Faktor, der die Größe des Suchraumes beeinflusst, ist die Frequenz mit welcher die Bildinformation geliefert wird. Dabei gilt je höher die Frequenz, desto kleiner kann man den Suchraum wählen, da bekannt ist, dass sich der aktuelle Frame F_i

im Vergleich zum vorhergegangen Frame F_{i-1} nur minimal geändert hat.

In Zukunft sind also entsprechende Optimierungen für das folgende System zusätzlich realisierbar und sollten als gültige Verbesserungen berücksichtigt werden.

Abschnitt 3.1 definiert die Begriffe des Intra-Matching und Inter-Matching im Zusammenhang mit dem Tracking von Punktmengen. In Abschnitt 3.2 wird ein naiver Ansatz präsentiert, der die Umsetzung des Intra- / Inter-Matching beschreibt. Abschnitt 3.3 beschreibt eine Umsetzung, die sich für Echtzeitanwendungen anbieten könnte, was die Detektion von Merkmalen und deren Zuordnung angeht. Eine mögliche Erweiterung wird in Abschnitt 3.5 vorgestellt. In allen Fällen führt das Tracking von korrespondierenden 2D-Punktmengen zu 3D-rekonstruierbaren Weltpunkten. Abschnitt 3.6 gibt eine Abgrenzung gegenüber zwei weiteren Verfahren.

3.1 Definition: Intra-Matching und Inter-Matching

Punktmenge Θ wird durch einen der Extraktoren aus Abschnitt 2.2 gewonnen. Dabei wird das Bild in Kacheln aufgeteilt. Die Kachelgröße richtet sich dabei nach der Bilddimension. So liefert z. B. ein 1024×1024 großes Bild mit einer Kachelgröße von 30 maximal 1156 Punkte. Pro Kachel wird immer das stärkste Feature ausgewählt, so dass man davon ausgehen kann, dass dieses Merkmal sehr eindeutig ist. Dieses Merkmal lässt sich gut durch einen der lokalen Matching-Metriken aus Unterabschnitt 2.3.1 beschreiben. Beim Anwenden des SIFT-Operators fällt die Beschreibung eines Merkmals jedoch weg, da die interne Transformation bereits einen Deskriptor hinzufügt.

Intra-Matching beschreibt ganz allgemein das Matching einer Punktmenge Θ zwischen zwei Stereobildern I_1 und I_2 zum Zeitpunkt t_i. Bei Berücksichtigung der auf Seite 40 genannten Einschränkungen führt dies zu erheblich verkleinerten Suchräumen bei der Korrespondenzbestimmung. Dies führt zu einer massiven Reduzierung an Berechnungen und senkt die Rechenzeit stark.

Inter-Matching beschreibt das Matching einer Punktmenge Θ zwischen Zeitpunkt t_i und t_{i+1}. Der Suchraum vergrößert sich dabei wesentlich, da bestimmte Einschränkungen nicht mehr gegeben sind. Vor allem die unbekannte Bewegungsrichtung des Systems lässt keine Einschränkung des Suchraumes zu. In Abbildung 3.1 wird der genaue Vorgang des Matching zwischen zwei aufeinanderfolgenden Bildpaaren dargestellt. Der Ansatz ist dabei auf mehrere Frames erweiterbar in Abhängigkeit von der Bewegungsstärke und der damit verbunden Größe der aktuellen Punktmenge Θ, die dann zu einer Re-Initialisierung von Θ nach einer gewünschten Anzahl an Frames führt, um weiterhin ein Tracking auf der Punktmenge durchzuführen.

Re-Initialisierung von Θ Veranlasst das Zurücksetzen der Punktmenge Θ durch eine erneute Merkmalsextraktion auf einem Bild aus der Bildfolge. Diese neue Menge dient als Basis für folgende Matchingphasen. Unter Umständen kann es sonst zu Problemen beim Matching kommen, da nicht ausreichend Punkte vorhanden sind, um diese auf Korrespondenz hin zu überprüfen und somit eine spätere Positionsbestimmung unmöglich wird.

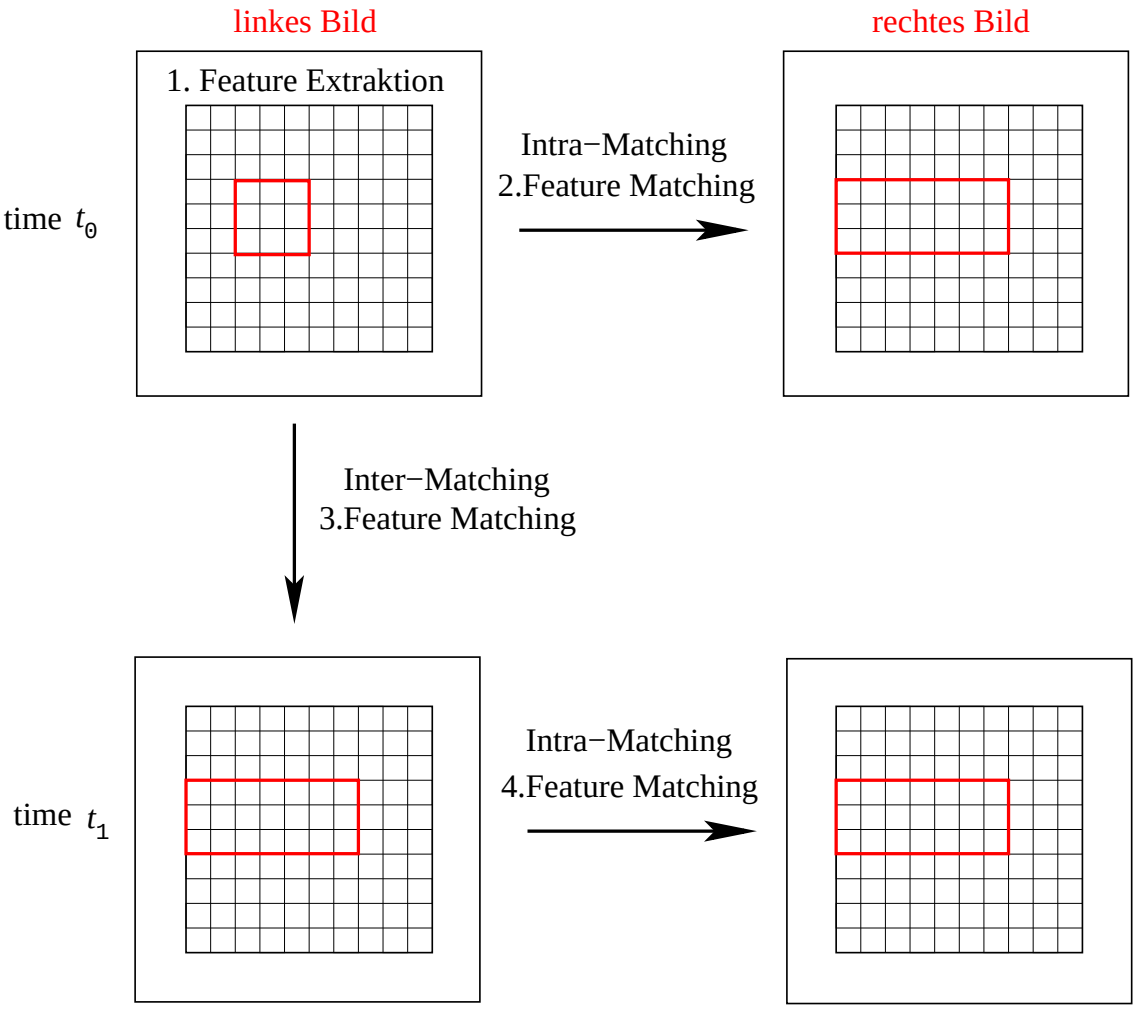
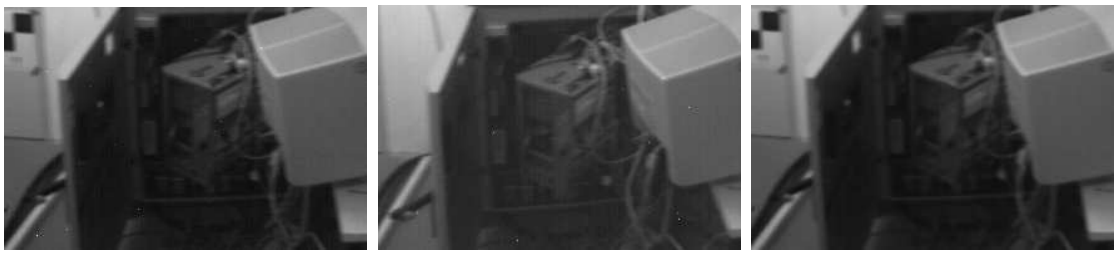


Bild 3.1: Vorgehensweise beim Matching zwischen 2 aufeinander folgenden Stereobildpaaren $P^1_{(I_1,I_2)}, P^2_{(I_1,I_2)}$: 1.) liefert die zu matchende Punktmenge, 2.) Intra-Matching für P^1, 3.) zeitversetztes Inter-Matching zwischen 2 Bildern: $P^1_{I_1}, P^2_{I_1}$, 4.) Intra-Matching auf Stereobildpaar P^2 auf bisher bestätigten Punkten aus 2. und 3. — rote Boxen stellen das zu findende lokale Pattern eines Punktes $p \in \Theta$ dar, plus die Shiftdimension für den Suchbereich.



(a) (b) (c)

Bild 3.2: Pixelrauschen: linkes Bild (a) vs. rechtes Bild (b); geglättetes linkes Bild (c)

3.2 Klassisches Tracking mit Intra- / Inter-Matching

Vor Beginn der Merkmalsextraktion müssen die Eingangsbilder geglättet werden. Dabei kann zwischen einem Medianfilter und einem Binomialfilter gewählt werden. Aufgrund von Pixelrauschen, das in den Aufnahmen auftritt, liefert der Medianfilter die besseren Vorverarbeitungsergebnisse. In Abbildung 3.2 ist ein Ausschnitt mit einem solchen Pixelrauschen dargestellt. Diese Artefakte stören sonst das Ergebnis der Extraktionsphase und erzeugen markante Punkte, welche im korrespondierenden Bildpaar nicht vorhanden sind und erhöhen dadurch nur unnötig die Anzahl der Berechnungen und potentielle Fehlzuordnungen, die die Schätzung negativ beeinflussen.

Bei naivem Vorgehen ergeben sich für die Testdaten auf dem realen Datensatz $B4$ Suchraumdimensionen von 151×61 für das Intra-Matching und Suchräume von 225×91 für jeden Inter-Matchingvorgang zwischen zwei Stereobildpaaren bei einer Bildgröße von 1024×1024 und einer Aufnahmefrequenz von 1Hz. Die Dimension des Fenster zur Bestimmung der Korrelation liegt bei 15×15. Daraus ergibt sich eine totale Anzahl an Berechnungen von $2\,072\,475$ für das Intra-Matching und $4\,606\,875$ für das Inter-Matching pro Feature, dessen Korrespondenz bestimmt werden soll. Diese großen Suchräume resultieren aus der nicht vorhandenen Kenntnis von Geometrie und Bewegung des Stereosystems, so dass besonders die Rechenzeit sehr stark ansteigt. Außerdem werden mehr Fehlzuordnugen „begünstigt“ und belasten das Ergebnis zusätzlich negativ.

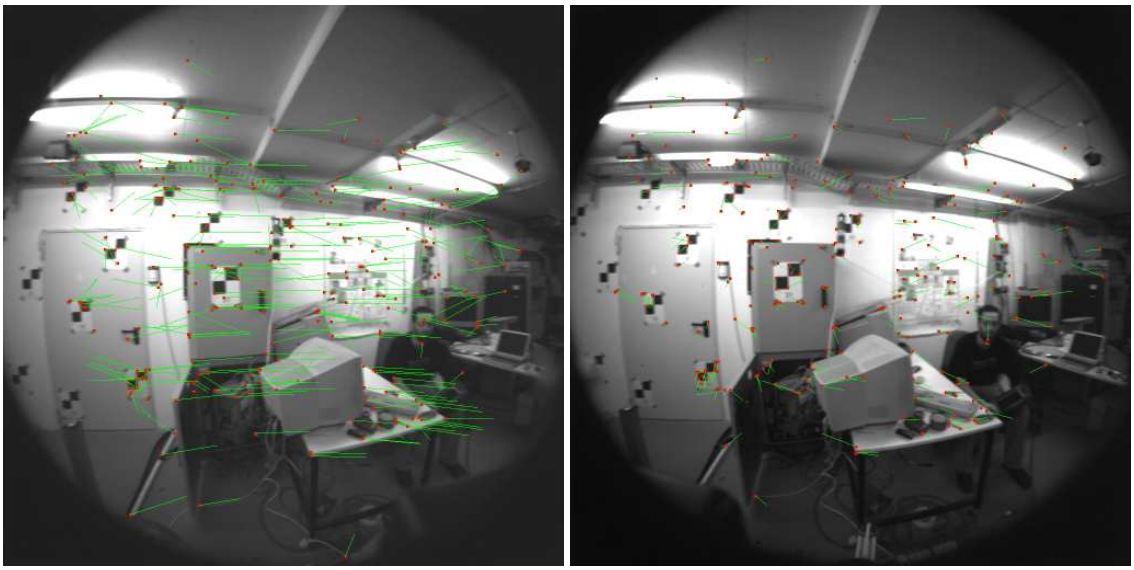


Bild 3.3: Originalbild: Matchingergebnisse mit KLT+Census nach dem letztem Intra-Matching. Zu sehen sind die Punkte, die in allen Bildern akzeptiert wurden. Die Trajektorien lassen extreme Fehlzuordnungen sofort erkennen. Linker Frame 4036 und rechter Frame 3968.

3.3 Multiresolution Tracking mit Intra- / Inter-Matching

Mit Hilfe einer Bildpyramide wie in Abbildung 3.4 wird nun versucht die Suchräume erheblich zu verkleinern. Unter der Annahme, das Features im Originalbild auch weiterhin Features im kleinskalierten Bild entsprechen. Zuerst wird das Originalbild mit einem gewählten Faktor kleinskaliert. Wir geben hier einen Pyramidenlevel von 4 an. Daraus errechnet sich der Faktor mit welchem das Bild verkleinert wird. In Tabelle 3.1 sind die jeweiligen Pyramidenstufen und deren Faktoren dargestellt um die gewünschte Anzahl an Pyramidenstufen zu erhalten. Für den hier betrachtenden Anwendungsfall ergibt sich also eine $8-$fache Verkleinerung des Originalbildes. Auf der untersten Ebene der Bildpyramide wird nun die Idee aus Abschnitt 3.2 übernommen und ein Intra-Matching auf das ebenfalls kleinskalierte rechte Kamerabild I_2 angewendet, nachdem bereits eine Punktmenge Θ aus Bild I_1 extrahiert worden ist. Die Korrespondenzanalyse in der Bildpyramide, wie diese in

Bild 3.4: Die Pyramidenstufen des linken Stereobildes aus Abbildung 2.3 mit Auflösungsstufen von 512×512, 256×256 und 128×128.

Abbildung 3.5 dargestellt wird, hat den Vorteil, dass durch das mehrmalige gegeneinander laufende Matching, die akzeptierten Korrespondenzen bereits mehrfach, über mehrere Stufen hinweg, bestätigt werden und somit eine größere Genauigkeit erwartet werden kann. Genaue Ergebnisse werden in Kapitel 4 aufgeführt.

Beim ersten Intra-Matching in der tiefsten Pyramidenstufe liegt ein großer Suchraum vor, der aber entsprechend dem Pyramidenlevel mit dem gleichen Faktor kleinskaliert wird. So ergibt sich hier beispielsweise ein Suchraum von $\left\lfloor \frac{151}{2^4} \right\rfloor \times \left\lfloor \frac{61}{2^4} \right\rfloor$ bei Übernahme der Startgröße aus Abschnitt 3.2. Weiterhin erfolgt eine enorme Reduzierung der Suchraumgröße für alle weiteren Zuordnungschritte durch den ebenfalls angepassten Suchraum in Abhängigkeit zur Pyramidentiefe.

Um das Matching des jeweils gleichen Features über mehrere Pyramidenstufen hinweg zu garantieren, muss ein Fine-Tuning des bestätigten Features in dessen direkter Nachbarschaft seiner hochskalierten Koordinate erfolgen. Dadurch bleibt beim Matching in den Zwischenstufen der Bildpyramide das Feature konstant und „läuft“ nicht weg vom

Pyramidenlevel	Faktor
1	2^0
2	2^1
3	2^2
4	2^3
...	...
n	$2^{(n-1)}$

Tabelle 3.1: Berechnung des jeweiligen Pyramidenlevel

ursprünglichen Merkmal. Das Fein-Tuning wird mit Hilfe des Moravec Operators umgesetzt. Es ergibt sich eine Größe von 5×5 für das Fein-Tuning der Features, in welchem der Moravec-Operator angewendet wird. Die Wahl fällt auf den Moravec Operator, da dessen Berechnung schneller erfolgt als jene von Harris-Corer und KLT. Im Gegensatz zur Definition in Unterabschnitt 2.2.1 wird das Minimum der Differenzen hier nur aus einem 3×3 Aussschnitt berechnet. Die Merkmale entsprechen aber weiterhin den initialen Kriterien von KLT oder Harris-Corner.

Für den dazugehörigen Matchingprozess genügt ebenfalls ein Suchfenster von 3×3 bezogen auf die bereits bekannte Koordinate des vorausgegangenen Matchings der darunter liegenden Pyramidenstufe. Angenommen ein Feature mit der Koordinate $(54, 33)$ aus ${I_1}^{py=4}$ wird auf $(27, 33)$ in ${I_2}^{py=4}$ gematcht. Dann reicht es aus in der darüber liegenden Pyramidenstufe ${I_1}^{py=3}$ das Feature in einer kleinen lokalen Nachbarschaft 3×3 um $(54 * 2, 33 * 2)$ anzupassen und dessen Korrespondenz in Bild ${I_2}^{py=3}$ um die vorgegebene Koordinate $(27 * 2, 33 * 2)$ zu suchen. Man vermutet also, dass sich das Feature nun in der Region um die Koordinaten $([107 - 109], [65 - 67])$ befindet und dessen Korrespondenz im Bereich von $([53 - 55], [65 - 67])$ liegt. Die Bildpyramide hat weiterhin den Vorteil, dass die Bildvorverarbeitung mit einem Glättungsfilter nicht mehr notwendig ist, da die Artefakte des Pixelrauschens in den unteren Stufen der Pyramide nicht mehr auftreten, was eine zusätzliche Ersparnis an Rechenzeit bringt. Der minimale Mehraufwand für die Skalierung wird dabei in Kauf genommen.

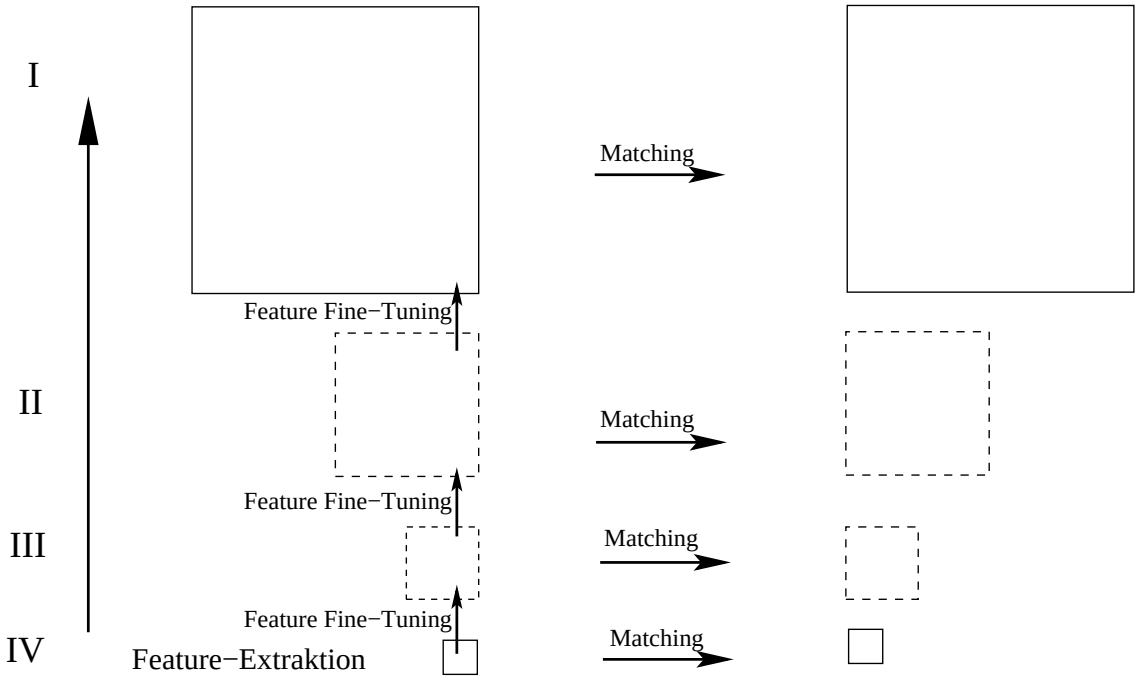


Bild 3.5: Bildpyramide mit 4 Stufen und den einzelnen Zwischenschritten für das Fein-Tuning und Matching zwischen zwei Bildern I_1, I_2 zu Zeitpunkt t_i: IV $\hat{=}$ kleinstes Bild, I $\hat{=}$ Originalbild. Der Ablauf geschieht in der sich wiederholenden Reihenfolge bis alle Stufen in der Pyramide abgearbeitet wurden: Skalierung $\mapsto$ Intra-Matching $\mapsto$ Fine-Tuning

3.4 SIFT-Operator und Intra- / Inter-Matching

Das Intra- / Inter-Matching mit dem SIFT-Operator funktioniert ein wenig anders. Statt der einmaligen Extraktion einer Punktmenge Θ wird nun für jedes Bild in einem Satz aus 2 Stereobildpaaren eine Merkmalsdetektion durchgeführt. In der Summe also für 4 Bilder. Übertragen auf das Intra-Matching bedeutet das, dass 2 Listen mit einer gewissen Anzahl an Merkmalen gegeneinander auf Korrespondenz hin untersucht werden. Die Ähnlichkeit zwischen Merkmalen wird mit Hilfe der euklidischen Distanz bestimmt, die sich aus den Deskriptordaten berechnen lässt. Das beste Paar aus den beiden Listen wird als korrespondierend betrachtet und beschreibt somit den gemeinsamen Weltpunkt. Dies wird für alle Merkmale in der Liste wiederholt, so dass jedem Merkmal aus Liste 1 ein Merkmal in Li-

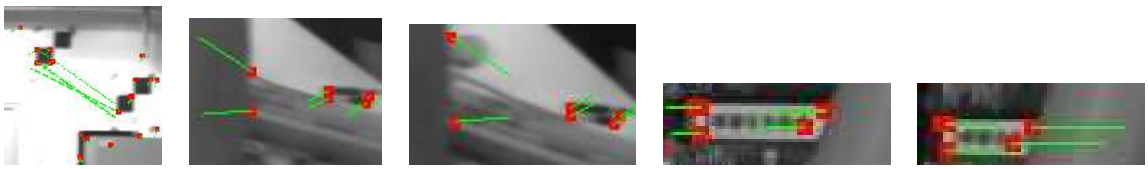
ste 2 zugeordnet werden kann, sofern die Schwelle von 150 nicht überschritten wird. Der gleiche Prozess wird in die entgegengesetzte Richtung erneut angewendet, so dass nur „verbundene“ Paare von Merkmalen zur späteren Schätzung herangezogen werden. Die bestätigten Korrespondenzen aus Liste 1 werden daraufhin beim Inter-Matching mit einer neuen Liste 3 verglichen. Ebenso erfolgt eine Korrespondenzanalyse zwischen Liste 3 und Liste 4, die jeweils die Koordinaten von Merkmaln des zweiten Stereopaars beinhalten.

3.5 Erweiterung zu Intra- / Inter-Rematching

Um nochmals die Genauigkeit von bisher erkannten Korrespondenzen zu erhöhen, wird außerdem ein Rematching von Punkten eingeführt. Damit nun ein Feature akzeptiert wird, muss es zusätzlich jeweils ein Intra- / Inter-Matching in die entgegen gesetzte Richtung bestehen. Das bedeutet, dass es auf seinen bisherigen Partner zurück gematcht werden muss. Dabei kann man verschiedene Toleranzen bezüglich der Exaktheit beim Rematching angeben. Von ganz streng, die keine Pixelabweichung erlaubt bis zur einer erlaubten Abweichung an Pixeln in der Nachbarschaft. Auf den ersten Blick liegt die Vermutung Nahe, dass die Rechenkosten wieder steigen könnten. Jedoch verringert das Rematching die Anzahl an Punkten, deren Genauigkeit jedoch höher ist. Folglich sollten sich für den weiteren Verlauf die Aufrufe der Matching-Routinen reduzieren und somit zahlreiche, teure Berechnungen entfallen. Die Laufzeit wird sich vermutlich in ähnlichen Dimensionen, wie bei o. g. Ansatz bewegen. Weiterhin sollten fehlerhafte Matchingergebnisse wie in Abbildung 3.6 zu sehen, vermieden werden können.

3.6 Abgrenzung zu anderen Arbeiten

Im Gegensatz zu [NBN06] werden hier mehrere Kombinationen von Feature Operatoren und Matching-Metriken angewandt, um so eine bestmögliche Kombination zu finden. Nister verwendet als Punktdetektor den Harris-Corner Operator und zur Korrespondenzfindung die normalisierte Kreuzkorrelation (NCC) mit einem 11×11 Fenster als Tracking-Kombination. Die Feature Extraktion findet auf beiden Bildern statt. Den Features wird



(a) (b) (c) (d) (e)

Bild 3.6: [v.l.n.r.] Beispiele für fehlerhaftes Matching durch: selbstähnliche Strukturen (a), Verdeckungen (b+c) und beide Fälle kombiniert (d+e). (a-c) aus Abbildung 2.3 und (d,e) aus Abbildung 2.9

jeweils der Wert seiner Nachbarschaft, beschrieben durch die NCC, zugeordnet. Akzeptiert werden nur Matches die auch zurück gematcht werden können. Das heißt Features mit einem ähnlichen NCC-Wert. Er verfolgt damit einen ähnlichen Ansatz wie Lowe mit dem SIFT-Operator, in dem nur detektierte Features in beiden Bildern miteinander verglichen werden. An Hand der Beschreibung werden nun Korrespondenzen gefunden, sofern eine gewisse Ähnlickeit gegeben ist. Damit wird versucht die Punktmenge bereits vor der Positionsschätzung zu stabilisieren. Aufgrund der geringen Bilddimension von 720×240 verzichtet er jedoch auf einen Pyramidenansatz. Ebenso verzichtet er auf Subpixelgenauigkeit.

Als Startgröße für das Disparitätslimit schwanken die Werte zwischen 3% und 30% der Bildgröße, je nach Aufnahmegeschwindigkeit und Beschaffenheit der Eingangsdaten. Im Normalfall liegt der Werrt jedoch bei 10%.

Die Größe einer Punktmenge Θ kann bis zu 5000 Features betragen, welche im Bild auf 10×10 Eimern verteilt sind. In jedem Eimer befinden sich 100 Features. Ein Feature liegt vor, wenn dessen Eckenstärke größer ist als die Ergebnisse anderer Kandidatenpixeln in einer 5×5 Nachbarschaft. Aus diesen detektierten Merkmalen in einem Eimer wird das stärkste Feature mit dem Quickselect-Algorithmus bestimmt.

Desweiteren sind die einzelnen Programmteile auf Intel MMX Instruktionen zugeschnitten und entsprechend optimiert. Damit werden bis zu 13 Frames pro Sekunde an Daten

verarbeitet, so dass die Bedingungen einer Echtzeit-Anwendung erfüllt sind.

Auch Hirschmüller [Hir03] benutzt den Harris-Corner Operator zur Merkmalsextraktion. Als Matching Metrik verwendet er eine adpative Variante der Absolutsumme (SAD), die mit multiplen Fenster arbeitet. Besonders an Objektgrenzen wird das Problem von Verdeckungen gelöst und führt zu weniger Fehlzuordnungen und das Erkennen von Tiefenunstetigkeiten. Das Stereo-Matching wird auf kalibrierten und rektifizierten Bildpaaren ausgeführt, so dass die Vorteile der Epipolargeometrie beim Suchen von Punktpaaren gültig sind und die Zuordnung vereinfachen. Die Aufnahmerate liegt bei 8 Frames pro Sekunde bei einer Bilddimension von 320×240.

Im hier vorgestellten Ansatz wird erstmals versucht die Rank- sowie die Census-Transformation als Metrik zur Verfolgung von Punktmengen zu nutzen. Beide wurden bisher nur zum reinen Stereo-Matching eingesetzt, die besondere Vorteile an Objektgrenzen besitzen, da die lokale Reihenfolge für die Transformation und nicht die Pixelintensitäten entscheidend ist. Wie bereits in 2.3.1 angedeutet, entfällt jedoch die Rank-Transformation als solche Metrik, da der Informationsverlust zu hoch ist. Die Einschränkung des Suchraumes ist zu ungenau für das Inter-Matching zwischen zwei Zeitpunkten t_i, t_{i+1}. Die Ergebnisse für die Census-Transformation werden detailliert in Abschnitt 4.3 präsentiert. Weiterhin wird auf sehr großen Bilddimensionen im Vergleich zu Nister und Hirschmüller gearbeitet, die die Anwendung eines Multiresolutionansatzes erfordert. Alle anderen Matching Metriken, die hier zur Korrespondenzermittlung benutzt werden, wurden schon mehrfach in der Literatur untersucht und eingesetzt. Sie dienen hier vor allem dazu einen Vergleich mit der Census-Transformation zu ermöglichen. Für exakte Ergebnisse der anderen Matching-Metriken und deren Qualität sei hier auf die Analyse von Aschwanden und Guggenbuhl [AG92] verwiesen.

Kapitel 4

Experimente und Ergebnisse

In diesem Kapitel erfolgt die Analyse der in Kapitel 2 vorgestellten und in Kapitel 3 verwendeten Algorithmen. Zur Auswertung kommen mehrere synthetische als auch reale Datensätze, die jeweils verschiedenen Kameratransformationen unterliegen, wobei die realen Aufnahmen nur eine exemplarische Auswertung ermöglichen, da bei Abschluss der Arbeit noch keine Kamerakalibrierung erfolgen konnte, so dass eine Schätzung von Lage und Position nicht möglich war. Die Anwendbarkeit der Operatoren in der Realität lässt sich jedoch recht gut an Hand von eingezeichneten Trajektorien erkennen. Sie geben ein Gefühl dafür wie robust das jeweilige Matching ist. Es lassen sich mögliche Fehlzuordnungen in der Regel an Hand konträrer Trajektorien eindeutig erkennen.

Abschnitt 4.1 beschreibt den Aufbau des Stereosystems, sowohl für die realen Aufnahmen als auch die Eingangsgrößen für die synthetisch generierten Stereodaten. In Abschnitt 4.2 werden die Versuchsparameter beschrieben. Abschnitt 4.3 präsentiert die Ergebnisse der Versuche. Abschließend wird in Abschnitt 4.4 das Resultat für die synthetischen Daten an Hand der vorher definierten Bewegung verifiziert.

Kamera:	Dalsa 1M28-SA (CMOS. 1024×1024 $10.6\mu m$ pitch)
Objektiv:	Pentax C418DX - TH mit Brennweite $f = 4.8mm$

Tabelle 4.1: Kameraparameter des Stereosystems

4.1 Versuchsaufbau

4.1.1 Reales Stereosystem

Zum Einsatz kommen zwei baugleiche Sensortypen. Dabei wird ein Öffnungswinkel von $97°$ erreicht. Die effektive Bilddimension beträgt $\left(1024 * \sqrt{2} \times 1024 * \sqrt{2}\right)$ Pixel. Abbildung 2.3 zeigt nochmals die damit gewonnenen Aufnahmen. Die Herstellerangaben für die Kameras sind in Tabelle 4.1 gelistet.

4.1.2 Synthetisches Stereosystem

Es werden drei synthetische Szenen ausgewertet. Das Stereosystem befindet sich dabei exakt in der Mitte des Raumes mit einer Baseline von $16cm$. Für alle Szenen gilt, dass die Texturdaten im Raum mit der Panoramakamera des DLR gewonnen wurden. Dabei handelt es sich um eine Zeilenkamera, die mit einem Laserscanner kombiniert ist und fusionierte Bilddaten liefert [STS05]. Die Texturen werden in OpenGL auf einen Quad abgebildet, so dass die o. g. Raumstruktur entsteht. Die Szene wird mit einer vorher definierten Bewegung gerendert.

Der Datensatz (I) wurde auf einem Pentium 4 mit 1.5 GHz und 1 GB Arbeitsspeicher unter Windows XP ausgewertet. Datensatz $(II + III)$ wurden auf einem Pentium Mobile-Centrino mit 1.5 GHz und 512 MB Arbeitsspeicher unter Linux ausgewertet.

Datensatz (I) Die simulierte Stereoszene wurde mit einer effektiven Bildgröße von 724×724 Pixel gerendert. Der Öffnungswinkel beträgt $56°$. Abbildung 4.1 zeigt ein Ste-

Bild 4.1: gerenderte Stereoszene aus Datensatz (I) mit einem Ausschnitt von Schloss Neuschwanstein: Frame 53. Erstellt aus Daten der DLR-Panoramakamera

reopaar aus jenem Datensatz. Die Größe des simulierten Raumes ist:

$$h = b = 8m,\ t = 12m \tag{4.1}$$

Datensatz (II) Die zweite Szene hat ebenfalls eine effektive Bildgröße von 724×724 Pixel. Der Öffnungswinkel beträgt allerdings $97°$ um auch einen Bezug zum realen Aufbau des System herstellen zu können. Die Größe des simulierten Raumes ist:

$$h = b = t = 4m \tag{4.2}$$

Datensatz (III) Die dritte Szene besitzt eine effektiven Bildgröße von 724×724 Pixel. Der Öffnungswinkel beträgt $97°$. Die Kamera befindet sich nicht in der Raummitte, sondern $5.5m$ von der Wand entfernt. Die Größe des simulierten Raumes ist:

$$h = b = t = 8m \tag{4.3}$$

Datensatz	(I)		$(II + III)$	
adaptive Parameter	Original	Pyramidenstufe $3 \rightarrow 2^2$	Original	Pyramidenstufe $3 \rightarrow 2^2$
Kachelgröße	30	7 bzw. 5	30	7 bzw. 5
Suchfenster	(81×11)	(21×3)	(101×21)	(25×5)
Korrelationsfenster	(15×15)	(15×15)	(15×15)	(15×15)
NCC	0.25	0.25	0.2	0.2
SAD	0.07	0.07	0.1	0.1
SSD	0.07	0.07	0.1	0.1
NSSD	0.10	0.10	0.1	0.1
CENSUS	0.18	0.18	0.2	0.2

Tabelle 4.2: Versuchsparameter für synthetische Datensätze. Die reduzierte Kachelgröße beim Tracking über 2 Frames erlaubt eine erhöhte initiale Featureanzahl und führt zu einer stabileren Messung. Die Werte für das jeweilige Matching-Verfahren geben die erlaubte prozentuale Abweichung gegenüber einer idealen Korrespondenz an.

4.2 Versuchsdurchführung

Die synthetischen Bilder wurden unter verschiedenen Bedingungen getestet. Die Datensätze (I) und (II) enthalten eine $360°$ Rotation in $5°$ Schritten um die y-Achse im jeweiligen Raum. In Datensatz (III) wird eine Translation à $0.1m$ Schritten pro Frame simuliert. Die Ergebnisse stehen für eine zurück gelegte Gesamtdistanz von $3.6m$. Die Parameter für die Größe des Suchfensters, die Dimension des Korrelationsfenster, die Matching-Toleranz, die Kachelgröße sind in Tabelle 4.2 angegeben. Das Tracking für Datensatz $(I + II)$ wurde jeweils über 72 Frames ausgeführt, wobei die Re-Initialisierung der Features einmal nach 1 Frame und beim zweiten Durchlauf erst nach 2 Frames erfolgte. Datensatz (III) besteht aus 36 Frames und versucht außerdem eine Bestimmung der Position für das System über mehrere Frames.

4.3 Ergebnisse

In Unterabschnitt 4.3.1 wird die Entwicklung von Punktmengen für ein Stereobildpaar aus einem realen Datensatz quantitativ bestimmt, um einen ersten Eindruck zu vermitteln in

		Matching Metrik				
Extraktor	*init*	NCC	SAD	SSD	NSSD	CENSUS
MORAVEC	363	330, 302, 294	336, 307, 305	330, 276, 274	341, 322, 321	341, 320, 313
KLT	384	338, 310, 299	352, 313, 307	347, 287, 280	357, 339, 336	357, 338, 328
HARRIS	385	336, 308, 296	355, 317, 310	348, 293, 286	357, 340, 336	357, 340, 330
		Rechendauer				
MORAVEC	363	51,16 sec	35,38 sec	31,83 sec	54,00 sec	40,84 sec
KLT	384	52,90 sec	38,01 sec	34,37 sec	57,32 sec	43,46 sec
HARRIS	385	52,83 sec	37,80 sec	34,15 sec	57,28 sec	44,05 sec

Tabelle 4.3: Originalbild: Anzahl lebender Features während eines Intra- / Inter-Matching auf 2 Stereobildpaaren und Gesamtrechenzeit auf Pentium Centrino 1.5 GHz mit 512 MB-Arbeitsspeicher unter Linux. Die Entwicklung entspricht dabei folgender Reihenfolge: Extraktion (*init*) $\mapsto$ Intra-Matching $\mapsto$ Inter-Matching $\mapsto$ Intra-Matching.

welchen Größendimensionen sich das Zuordnen von Punktmengen abspielt. Die Simulationsdaten werden in Unterabschnitt 4.3.2 ausgewertet. Die geschätzten Daten erlauben eine qualitative Aussage bezüglich der definierten Bewegung durch einen direkten Vergleich mit den Idealwerten.

4.3.1 Feature-Entwicklung

Die hier vorgestellten Ergebnisse in Tabelle 4.3 (ohne Bildpyramide) und 4.4 (mit Bildpyramide) stellen die Lebensdauer beim Intra- / Inter-Matching dar, wie es in Abschnitt 4.1 beschrieben wird. Die Ergebnisse stammen aus dem realen Testdatensatz $B4$ (s. Tabelle A.2), der synchronisierten Frames $F3967$ des rechten Bildes und $F4035$ des linken Bildes. Abbildung 4.2 zeigt die linken Kamerabilder des jeweiligen Stereopaars mit eingezeichneten Bewegungstrajektorien. Die Werte geben eine Auskunft darüber wieviel von den initial extrahierten Features nach mehreren Matching-Vorgängen noch gültig und somit in all 4 Bildern vorhanden sind. Der letzte Wert in jeder Zelle entspricht genau dieser Information bezüglich der aktuellen Punktmenge Θ, die für das nächste Inter-Matching dienen würde bzw. aus welcher eine relative Positionsbestimmung möglich ist. Wie zu erwarten liefern alle Operatoren kombiniert mit den jeweiligen Matching Metriken vergleichbare Größen-

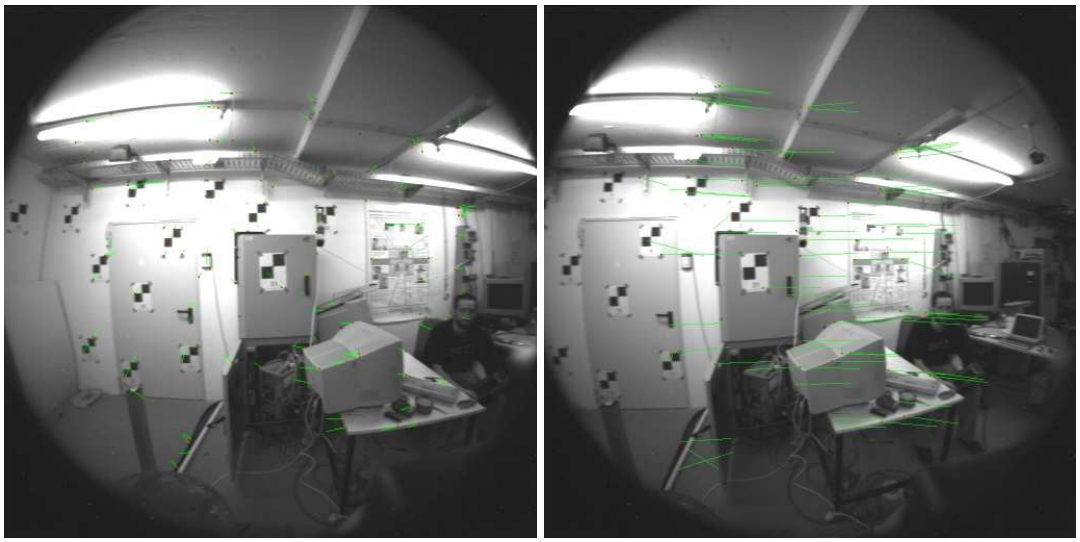
		Matching Metrik				
Extraktor	*init*	NCC	SAD	SSD	NSSD	CENSUS
MORAVEC	224:	125, 94, 75	155, 110, 90	130, 66, 53	168, 150, 126	130, 99, 73
KLT	363:	147, 107, 84	189, 122, 97	143, 68, 52	211, 190, 154	141, 102, 74
HARRIS	332:	136, 98, 77	175, 114, 90	135, 67, 51	200, 179, 145	126, 93, 70
		Rechendauer				
MORAVEC	224:	0,65 sec	0,40 sec	0,34 sec	0,61 sec	0,42 sec
KLT	363:	0,59 sec	0,48 sec	0,41 sec	0,72 sec	0,50 sec
HARRIS	332:	0,57 sec	0,48 sec	0,38 sec	0,70 sec	0,48 sec

Tabelle 4.4: Bildpyramide: Anzahl lebender Features während eines Intra- / Inter-Matching auf 2 Stereobildpaaren und Gesamtrechenzeit auf Pentium Centrino 1.5 GHz mit 512 MB-Arbeitsspeicher unter Linux. Die Entwicklung entspricht dabei folgender Reihenfolge: Extraktion (*init*) $\mapsto$ Intra-Matching $\mapsto$ Inter-Matching $\mapsto$ Intra-Matching.

ordnungen bezogen auf die Anzahl der Features und der damit verbundenen Rechenzeit. Die Größe von Θ hat entscheidenden Einfluss auf die Rechenzeit. So braucht die Census-Transformation deutlich mehr Zeit als z. B. SAD oder SSD, da diese die zugehörige Matching Routine um ein vielfaches häufiger aufrufen muss, obwohl die Transformation eines Pattern eine schnellere Berechnung vermuten lässt. Das sagt bisher aber noch nichts über die Qualität der akzeptierten Punktkorrespondenzen aus, deren qualitative Auswertung im nachfolgenden Unterabschnitt 4.3.2 erfolgt.

In Abbildung 4.2 werden nochmals exemplarisch die Ergebnisse des Inter-Matching für den realen Datensatz gezeigt. Als Kombination wurde KLT mit SAD bzw. Census gewählt unter Verwendung des Multiresolutionansatzes. Die eingezeichneten Trajektorien erlauben eine grobe quantitative Analyse der Matchingergebnissen, um diese auf Richtigkeit hin zu untersuchen. Die Trajektorien in den beiden links angeordneten Bildern lassen durch ihre Krümmung hier auch nochmals klar die Verzeichnung der Kamera erkennen. Sie repräsentieren in diesem Fall die akzeptierten Korrespondenzen aus der vorangegangenen Intra-Matching Phase. Bei entsprechender Entzerrung der Kamera können noch weitaus bessere Ergebnisse für jeden weiteren Schritt erwartet werden.

(a)



(b)

Bild 4.2: Die Bilder geben die Ergebnisse an Hand von Trajektorien wieder. Hier: Multiresolutionansatz, Inter-Matching Ergebnisse zwischen den linken Bildern von zwei Stereobildpaaren. Frames: 4035 und 4036 mit KLT+SAD (a), KLT+Census (b)

4.3.2 Lage- und Positionsschätzung

Für alle nachfolgenden Tabellen[1] gilt, dass sie im jeweils ersten Drittel den Mittelwert und die Standardabweichung bezüglich der geschätzten Bewegung angeben. Beim Tracking über 1 Frame entspricht dies einer Drehung um $5°$ und beim Tracking über 2 Frames einer Drehung um $10°$. Das zweite Drittel der Tabellen gibt den Mittelwert an, der zu Beginn der Schätzung vorliegenden Punkte, die aus dem Intra- / Inter-Matching hervorgehen. Außerdem wird der gemittelte Wert der gelöschten, inkonsistenten Punkte angegeben, die vom Schätzalgorithmus nicht berücksichtigt wurden. Bereits daraus lässt sich eine Aussage über die Qualität der gelieferten Punkte machen. Je weniger Punkte hier wegfallen, desto besser ist die gegebene Punktmenge Θ. Der letzte Tabellenabschnitt gibt sowohl die Gesamtrechenzeit für das Tracking über alle Frames an, sowie die Rechenzeit für die Korrespondenzbestimmung für genau einen Punkt an, berechnet aus der Relation von Gesamtpunktmenge und Gesamtrechenzeit.

Die Bewertung der Daten erfolgt im Anschluss an jede Tabelle. Sie ergibt sich dabei aus mehreren Faktoren. σ bestimmt die Gewichtung der jeweiligen Faktoren. Die Standardabweichung (STD) geht zu 8 Teilen, die Zeit (t) zu 4 Teilen, der Mittelwert (MW) zu 2 Teilen und die Anzahl der Features (ft) zu 1 Teil multipliziert mit der Platzierung m ein. Die Punktverteilung k für eine bestimmte Kombination ergibt sich also aus

$$k \;=\; \sum_{i=0}^{3}(80 - m * 5) * \sigma^i \quad \text{mit } m \in [1 \ldots 15]$$

und bestimmt die Kombination, welche das beste Resultat für die Schätzung ergibt.

Datensatz (I) Als beste Kombination für große Räume ergibt sich aus nachfolgenden Tabellen KLT und SAD. Die Standardabweichung beträgt knapp ein Grad, bei einer geringen Anzahl an Fehlkorrespondenzen und einer schnellen Berechnung. Die langsamen Rechenzeiten für SSD sind ungewöhnlich, da sie normalerweise im gleichen Bereich wie SAD liegen sollten. Die Ergebnisse aus Datensatz (II) bestätigen dies auch später. Dort haben beide in etwa dieselben Laufzeiten. Vermutlich liegt es an einer systemabhängigen Kompilierung des Codes (Windows vs. Linux). Im Normalfall bietet sich also SSD

[1] die Onlineversion des Dokuments enthält zusätzlich Auswertungen für den Standardansatz.

durchaus als alternative Matching-Metrik an. Die Laufzeiten von NSSD und NCC liegen aufgrund ihrer intensiveren Berechnung entsprechend höher. Die Ergebnisse der Census-Transformation liegen sowohl was die Laufzeit angeht, als auch die Qualität der gelieferten verfolgbaren Punkte hinter den Erwartungen. Die extrem hohen Standardabweichungen lassen eine praxisnahe Umsetzung vorerst nicht zu. Das Tracking kombiniert mit Rematching liefert ebenfalls keine stabileren Ergebnisse und bleibt hinter den Erwartungen bzgl. der Präsizion zurück, so dass der Mehraufwand nicht gerechtfertigt ist.

Der SIFT-Operator läuft nicht im direkten Vergleich, da zum einem die Extraktion von Merkmalen sehr zeitintensiv ist und zum anderem für jedes Bild im Intra- / Inter-Matching Prozess gemacht werden muss. Die Extraktionszeit ist abhängig von der jeweiligen Szene. Je mehr Punkte den initialen Test (s. Punkt 1 in Unterabschnitt 2.2.4) bestehen, desto höher ist entsprechend die Laufzeit. Die Extraktionszeit für die beiden Bilder in Abbildung 4.1 liegt bei 2.105 Sekunden auf einem Pentium-Centrino 1.5GHz. Es werden außerdem nur Listen von extrahierten Features gegenseitig auf Korrespondenz hin untersucht und arbeiten somit unabhängig von Parametern wie Suchfenster und Korrelationsfenster. Die erzielten Schätzergebnisse lassen deutlich erkennen, dass der SIFT-Operator in der Lage ist markante Punkte über mehrere Frames zu verfolgen. Vor allem bei extremen Bewegungen werden noch erstaunlich gute Punktmengen wiedergefunden. Zu den Laufzeiten lässt sich noch sagen, dass diese abfallend sind, da die Re-Initialisierungsphase beim Tracking über mehrere Frames seltener aufgerufen wird. Im Gegensatz zu den anderen Verfahren ist die Extraktion der Faktor, der maßgeblichen Einfluss auf die Performanz des Trackings hat und nicht die Matchingphasen. Tabelle 4.13 gibt einen genauen Überblick zu Laufzeiten und Qualität der Schätzergebnissen. Der Versuch auf reduzierten Auflösungen zu arbeiten, führt vorerst nur zu instabilen Ergebnissen wie Tabelle 4.14 zeigt und Bedarf noch weiterer Untersuchungen, da der Schätzalgorithmus unabhängig von der Bildgröße arbeitet. Normalerweise sollte also eine Bestimmung mit verbesserter Laufzeit möglich sein.

Mittelwert und Standardabweichung in Grad vom idealen Wert: $5°$

Extraktor	NCC	SAD	SSD	NSSD	CENSUS
HARRIS	4.7165; 1.2323	4.7256; 0.8833	4.7237; 1.0998	4.5797; 0.9135	5.8233; 8.3647
KLT	4.7564; 1.0633	4.7329; 0.8768	4.6787; 1.2616	4.6349; 1.0306	5.5490; 8.1592
MORAVEC	4.8595; 1.3875	4.8532; 1.1802	4.9320; 1.3105	4.7939; 0.9452	4.8550; 3.8464

Prozentuale Angabe der gelöschten Features im Mittel bezogen auf die Startpunktmenge

HARRIS	0.75%	1.25%	1.43%	6.95%	7.41%
KLT	0.74%	1.11%	1.28%	6.42%	7.08%
MORAVEC	0.99%	1.31%	1.15%	5.98%	5.39%

Rechenzeit in Sekunden (s) pro Feature(ft)

HARRIS	$0.004\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.014\frac{s}{ft}$	$0.006\frac{s}{ft}$	$0.004\frac{s}{ft}$
KLT	$0.004\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.014\frac{s}{ft}$	$0.006\frac{s}{ft}$	$0.004\frac{s}{ft}$
MORAVEC	$0.003\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.012\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.003\frac{s}{ft}$

	NCC	SAD	SSD	NSSD	CENSUS	bester Extraktor
HARRIS	590	970	485	645	195	2885
KLT	775	**1050**	375	600	270	**3070**
MORAVEC	605	740	465	735	500	3045
bester Matcher:	1970	**2760**	1325	1980	965	9000

Tabelle 4.6: Datensatz (I): Tracking Ergebnisse für eine $360°$ Rotation um die y-Achse über 72 Frames in $5°$ Schritten: ohne Rematching, mit Bildpyramide, mit Re-Initialisierung der Punktmenge nach 1 Frame. Die Werte repräsentieren den Mittelwert und die Standardabweichung in $°$ bzgl. des Idealwertes von $5°$, sowie Mittelwerte für die Startpunktmenge und die durchschnittliche Anzahl an gelöschten inkonsistenten Features.

Mittelwert und Standardabweichung in Grad vom idealen Wert: $5°$

Extraktor	NCC	SAD	SSD	NSSD	CENSUS
HARRIS	4.9325; 1.1527	4.9234; 0.9392	4.7928; 1.0382	4.7358; 1.0578	5.2834; 3.6018*
KLT	4.9239; 1.1244	4.9197; 0.8960	4.7486; 1.1441	4.8097; 0.9257	4.3614; 3.2325*
MORAVEC	4.9456; 1.2848	4.9544; 1.0134	4.7963; 1.0874	4.9164; 0.9576	3.7217; 4.5065

Prozentuale Angabe der gelöschten Features im Mittel bezogen auf die Startpunktmenge

HARRIS	0.93%	1.05%	1.20%	6.07%	7.88%
KLT	1.03%	0.97%	1.11%	5.84%	7.28%
MORAVEC	0.95%	1.11%	1.13%	5.07%	5.57%

Rechenzeit in Sekunden (s) pro Feature(ft)

HARRIS	$0.007\frac{s}{ft}$	$0.003\frac{s}{ft}$	$0.027\frac{s}{ft}$	$0.011\frac{s}{ft}$	$0.014\frac{s}{ft}$
KLT	$0.007\frac{s}{ft}$	$0.003\frac{s}{ft}$	$0.027\frac{s}{ft}$	$0.011\frac{s}{ft}$	$0.013\frac{s}{ft}$
MORAVEC	$0.006\frac{s}{ft}$	$0.003\frac{s}{ft}$	$0.022\frac{s}{ft}$	$0.010\frac{s}{ft}$	$0.009\frac{s}{ft}$

	NCC	SAD	SSD	NSSD	CENSUS	bester Extraktor
HARRIS	605	965	515	535	195	2815
KLT	680	**1025**	375	805	250	**3135**
MORAVEC	610	940	490	760	250	3050
bester Matcher:	1895	**2930**	1380	2100	695	9000

Tabelle 4.8: Datensatz (I): Tracking Ergebnisse für eine $360°$ Rotation um die y-Achse über 72 Frames in $5°$ Schritten: mit Rematching, mit Bildpyramide, mit Re-Initialisierung der Punktmenge nach 1 Frame. Die Werte repräsentieren den Mittelwert und die Standardabweichung in $°$ bzgl. des Idealwertes von $5°$, sowie Mittelwerte für die Startpunktmenge und die durchschnittliche Anzahl an gelöschten inkonsistenten Features.
* die Auswertung war nur bis Frame 36 möglich.

Mittelwert und Standardabweichung in Grad vom idealen Wert: 10°

Extraktor	NCC	SAD	SSD	NSSD	CENSUS
HARRIS	10.0883; 1.9306	10.3397; 1.3720	10.4417; 1.7059	10.2512; 1.3444	11.75; 17.25
KLT	10.2988; 1.6317	10.2360; 1.1968	10.9230; 1.6714	10.2814; 1.3025	14.46; 15.12
MORAVEC	9.6688; 2.0496	9.9183; 1.2573	9.9988; 1.9643	10.0428; 1.6613	8.01; 6.19

Prozentuale Angabe der gelöschten Features im Mittel bezogen auf die Startpunktmenge

HARRIS	1.48%	2.21%	2.38%	8.21%	8.72%
KLT	1.81%	2.03%	2.25%	7.89%	8.06%
MORAVEC	1.83%	2.24%	2.45%	7.38%	6.07%

Rechenzeit pro Feature

HARRIS	$0.003\frac{s}{ft}$	$0.001\frac{s}{ft}$	$0.013\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.004\frac{s}{ft}$
KLT	$0.003\frac{s}{ft}$	$0.001\frac{s}{ft}$	$0.013\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.004\frac{s}{ft}$
MORAVEC	$0.003\frac{s}{ft}$	$0.001\frac{s}{ft}$	$0.011\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.003\frac{s}{ft}$

	NCC	SAD	SSD	NSSD	CENSUS	bester Extraktor
HARRIS	655	915	390	655	215	2830
KLT	790	**1070**	445	710	260	**3275**
MORAVEC	495	1000	445	605	350	2895
bester Matcher:	1940	**2985**	1280	1970	825	9000

Tabelle 4.10: Datensatz (I): Tracking Ergebnisse für eine 360° Rotation um die y-Achse über 72 Frames in 10° Schritten: ohne Rematching, mit Bildpyramide, mit Re-Initialisierung der Punktmenge nach 2 Frames. Die Werte repräsentieren den Mittelwert und die Standardabweichung in $^\circ$ bzgl. des Idealwertes von 10°, sowie Mittelwerte für die Startpunktmenge und die durchschnittliche Anzahl an gelöschten inkonsistenten Features.

Mittelwert und Standardabweichung in Grad vom idealen Wert: 10°

Extraktor	NCC	SAD	SSD	NSSD	CENSUS
HARRIS	10.8086; 4.8949	10.6638; 4.6638	10.8222; 4.7848	9.7821; 1.3413	9.00; 26.47
KLT	10.5058; 2.8015	10.0295; 1.4385	10.1130; 1.5212	9.7496; 1.2840	12.70; 17.53
MORAVEC	10.7271; 2.0360	10.2998; 1.5081	10.4009; 1.9809	9.8420; 1.3062	14.29; 11.69

Prozentuale Angabe der gelöschten Features im Mittel bezogen auf die Startpunktmenge

HARRIS	1.90%	1.71%	2.44%	7.71%	7.62%
KLT	2.15%	1.71%	2.36%	7.16%	9.72%
MORAVEC	2.16%	1.86%	2.54%	6.50%	7.61%

Rechenzeit in Sekunden (s) pro Feature(ft)

HARRIS	$0.006\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.025\frac{s}{ft}$	$0.010\frac{s}{ft}$	$0.016\frac{s}{ft}$
KLT	$0.006\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.025\frac{s}{ft}$	$0.010\frac{s}{ft}$	$0.014\frac{s}{ft}$
MORAVEC	$0.005\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.022\frac{s}{ft}$	$0.009\frac{s}{ft}$	$0.014\frac{s}{ft}$

	NCC	SAD	SSD	NSSD	CENSUS	bester Extraktor
HARRIS	470	620	340	800	160	2390
KLT	635	**1005**	625	895	225	**3385**
MORAVEC	670	865	545	900	245	3225
bester Matcher:	1775	**2490**	1510	2595	630	9000

Tabelle 4.12: Datensatz (I): Tracking Ergebnisse für eine 360° Rotation um die y-Achse über 72 Frames in 10° Schritten: mit Rematching, mit Bildpyramide, mit Re-Initialisierung der Punktmenge nach 2 Frames. Die Werte repräsentieren den Mittelwert und die Standardabweichung in $^\circ$ bzgl. des Idealwertes von 10°, sowie Mittelwerte für die Startpunktmenge und die durchschnittliche Anzahl an gelöschten inkonsistenten Features.

SIFT - Operator

Rotationsschritte	MW und STD	gelöschte Feature in %	Rechenzeit
5°	5.0011; 0.4179	1.75%	679.4s
10°	9.8974; 0.7525	2.32%	494.1s
15°	15.5171; 1.3388	2.37%	417.5s
20°	19.8507; 1.0941	2.08%	376.4s
25°	24.7974; 4.1775	1.38%	333.7s
30°	31.9862; 7.0556	2.93%	323.0s
35°	35.1709; 3.4687	5.76%	296.1s

Tabelle 4.13: Datensatz (I): Tracking Ergebnisse für eine $360°$ Rotation um die y-Achse über 72 Frames für den SIFT-Operator. Die Re-Initialisierung erfolgt jeweils nach $1, 2, 3, 4, 5, 6,$ und 7 Frames. Stabile Features werden auch über große Distanzen wiedergefunden. Die Prozentangabe ergibt sich aus der Relation der durchschnittlichen Startpunktmenge und der im Mittel gelöschten Anzahl an Punkten, die nicht für die Schätzung herangezogen werden. Beispielsweise gibt es beim Tracking über $5°$ im Durchscnitt 394.5 Punkte von denen 6.9 gelöscht werden. Die Punktmengenangaben können aus dem dazugehörigen da-04.tex - Dokument extrahiert werden durch das Entkommentieren von 2 Zeilen zu Beginn des Dokumentes mit anschließendem Aufruf des Makefiles.

SIFT - Operator

Translationsschritte	MW und STD	gelöschte Feature in %	Rechenzeit
5 °	8.9364; 4.0018	1.83%	136.0s
10°	16.5360; 6.9351	2.21%	99.1
15°	26.4159; 8.1423	2.61%	86.4s

Tabelle 4.14: Die Tabelle zeigt die fehlerhaften Messungen, die aus korrespondierenden SIFT-Merkmalen auf einer reduzierten Auflösung von 362×362 entstanden sind. Normalerweise sollte die Schätzung auch auf reduziereten Bilddaten möglich sein, da diese unabhängig von der Bilddimension arbeitet. Wie erwartet ist die Rechenzeit schon enorm reduziert worden. Wodurch die Fehler bei der Schätzung entstehen ist z. Zt. unklar, da die Trajektorienbilder durchaus plausibel erscheinen.

Datensatz (II) Bei Betrachtung der Ergebnisse des Datensatzes (II) spricht vieles für den Einsatz aus einer Kombination von Harris-Corner und SAD. Im Gegensatz zu Datensatz (I), wo als beste Kombination KLT und SAD bestimmt werden konnte. Wie in Abschnitt 2.2 beschrieben ähneln sich beide in ihrer Herleitung aus der Hesse-Matrix und liefern somit auch qualitativ vergleichbare Ergebnisse, was die Extraktion angeht. So gesehen widerspricht sich das hier erhaltene Ergebnis nicht der vorher gewonnen Erkenntnis. Viel entscheidener ist die Bestätigung der Absolutsumme (SAD) als zuverlässige und schnelle Matching-Metrik. Die bestätigten Punktkorrespondenzen lassen auch in diesem Fall eine gute Schätzung von Lage und Position zu.

Nachdem die Ergebnisse des Rematching in Datensatz (I) keine Verbesserung herbeiführen konnten, erzielen sie auch diesmal nur teilweise die gewünschten Verbesserungen. Die Standardabweichung zwischen Harris-Corner und SAD mit $0.4638°$ und Harris-Corner, SAD und Rematching mit $0.4504°$ weist nur eine geringe Differenz auf, die die Schätzung nicht entscheidend (positiv/ negativ) beeinflusst. Die Entwicklung der Standardabweichung mit der Kombination aus KLT und SAD ist beispielsweise mit $0.4757°$ zu $0.4842°$ sogar wieder gegenläufig. Es scheinen eher zufällige Einflüsse des Rematching zu sein, die sich auf die Schätzung auswirken. Sowohl die längere Rechenzeit, als auch nur die minimal erhöhte Genauigkeit der Zuordnung fällt negativ ins Gewicht und rechtfertigt den zusätzlichen Aufwand nicht im Vergleich zum reinen Intra- / Inter-Matching . Die Resultate aus Datensatz (I) verdeutlichen zusätzlich, dass das Rematching die Schätzung in großen Räumen sogar negativ beeinflussen kann. Deswegen werden die Ergebnisse des Rematching für weitere Datensätze nicht mehr in Betracht gezogen.

Der SIFT-Operator erzielt auch hier wieder sehr gute Schätzergebnisse. Er ist in diesem Fall sogar in der Lage markante Punkte über eine noch größere Distanz (bis zu $45°$) zu verfolgen. Hier spielt sicherlich der vergrößerte Öffnungswinkel des Kamerasystems eine entscheidende Rolle. Die Laufzeiten verhalten sich vergleichbar zu denen in Datensatz (I) und sind aus gleichen Gründen ähnlich hoch. Einen genauen Überblick zu Laufzeiten und Qualität der Schätzergebnisse gibt Tabelle 4.23.

Mittelwert und Standardabweichung in Grad vom idealen Wert: $5°$

Extraktor	NCC	SAD	SSD	NSSD	CENSUS
HARRIS	4.9700; 0.4526	4.9552; 0.4833	4.9632; 0.4815	4.9943; 0.4762	5.4145; 6.7699
KLT	4.9596; 0.4757	4.9828; 0.5308	4.9828; 0.4943	5.0000; 0.5350	6.3509; 8.6226
MORAVEC	5.0262; 0.3936	5.0090; 0.4154	5.0216; 0.4269	5.0337; 0.4775	4.1054; 5.9925

Prozentuale Angabe der gelöschten Features im Mittel bezogen auf die Startpunktmenge

HARRIS	0.25%	0.20%	0.26%	0.34%	2.61%
KLT	0.30%	0.29%	0.31%	0.39%	2.37%
MORAVEC	2.59%	1.37%	1.67%	1.75%	1.62%

Rechenzeit in Sekunden (s) pro Feature(ft)

HARRIS	$0.005\frac{s}{ft}$	$0.003\frac{s}{ft}$	$0.003\frac{s}{ft}$	$0.004\frac{s}{ft}$	$0.005\frac{s}{ft}$
KLT	$0.005\frac{s}{ft}$	$0.003\frac{s}{ft}$	$0.003\frac{s}{ft}$	$0.004\frac{s}{ft}$	$0.005\frac{s}{ft}$
MORAVEC	$0.004\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.004\frac{s}{ft}$	$0.004\frac{s}{ft}$

	NCC	SAD	SSD	NSSD	CENSUS	bester Extraktor
HARRIS	690	635	645	745	135	2850
KLT	625	630	630	530	100	2515
MORAVEC	825	**1025**	925	590	270	**3635**
bester Matcher:	2140	**2290**	2200	1865	505	9000

Tabelle 4.16: Datensatz (II): Tracking Ergebnisse für eine $360°$ Rotation um die y-Achse über 72 Frames in $5°$ Schritten: ohne Rematching, mit Bildpyramide, mit Re-Initialisierung der Punktmenge nach 1 Frame. Die Werte repräsentieren den Mittelwert und die Standardabweichung in $°$ bzgl. des Idealwertes von $5°$, sowie Mittelwerte für die Startpunktmenge und die durchschnittliche Anzahl an gelöschten inkonsistenten Features.

Mittelwert und Standardabweichung in Grad vom idealen Wert: $5°$

Extraktor	NCC	SAD	SSD	NSSD	CENSUS
HARRIS	4.9579; 0.4641	4.9152; 0.4418	4.9432; 0.4455	4.9161; 0.4740	4.4408; 1.4259
KLT	4.9961; 0.4598	4.9471; 0.5114	4.9650; 0.5094	4.9449; 0.5078	4.5992; 1.3437
MORAVEC	4.9661; 0.4832	4.9806; 0.5336	4.9770; 0.5086	4.9753; 0.5415	4.3713; 4.4113

Prozentuale Angabe der gelöschten Features im Mittel bezogen auf die Startpunktmenge

HARRIS	0.41%	0.12%	0.20%	0.24%	3.42%
KLT	0.48%	0.12%	0.26%	0.24%	3.69%
MORAVEC	0.25%	0.14%	0.22%	0.21%	2.33%

Rechenzeit in Sekunden (s) pro Feature(ft)

HARRIS	$0.010\frac{s}{ft}$	$0.006\frac{s}{ft}$	$0.006\frac{s}{ft}$	$0.008\frac{s}{ft}$	$0.014\frac{s}{ft}$
KLT	$0.009\frac{s}{ft}$	$0.006\frac{s}{ft}$	$0.006\frac{s}{ft}$	$0.008\frac{s}{ft}$	$0.014\frac{s}{ft}$
MORAVEC	$0.008\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.008\frac{s}{ft}$	$0.012\frac{s}{ft}$

	NCC	SAD	SSD	NSSD	CENSUS	bester Extraktor
HARRIS	675	**950**	880	690	150	**3345**
KLT	790	655	630	655	175	2905
MORAVEC	665	705	780	475	125	2750
bester Matcher:	2130	**2310**	2290	1820	450	9000

Tabelle 4.18: Datensatz (II): Tracking Ergebnisse für eine $360°$ Rotation um die y-Achse über 72 Frames in $5°$ Schritten: mit Rematching, mit Bildpyramide, mit Re-Initialisierung der Punktmenge nach 1 Frame. Die Werte repräsentieren den Mittelwert und die Standardabweichung in $°$ bzgl. des Idealwertes von $5°$, sowie Mittelwerte für die Startpunktmenge und die durchschnittliche Anzahl an gelöschten inkonsistenten Features.

Mittelwert und Standardabweichung in Grad vom idealen Wert: 10°

Extraktor	NCC	SAD	SSD	NSSD	CENSUS
HARRIS	9.8600; 0.4321	9.8573; 0.4638	9.8234; 0.4546	9.7976; 0.5093	11.7966; 8.4667
KLT	9.7489; 0.4794	9.8627; 0.4754	9.8162; 0.4848	9.7686; 0.5709	9.6909; 11.3805
MORAVEC	9.7445; 0.5916	9.7516; 0.6594	9.7680; 0.5323	9.6922; 0.7567	11.1435; 5.7940

Prozentuale Angabe der gelöschten Features im Mittel bezogen auf die Startpunktmenge

HARRIS	1.08%	0.73%	0.87%	0.83%	3.23%
KLT	1.07%	0.68%	0.88%	0.84%	2.90%
MORAVEC	0.88%	0.69%	0.86%	0.84%	1.86%

Rechenzeit in Sekunden (s) pro Feature(ft)

HARRIS	$0.004\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.004\frac{s}{ft}$	$0.005\frac{s}{ft}$
KLT	$0.004\frac{s}{ft}$	$0.003\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.004\frac{s}{ft}$	$0.005\frac{s}{ft}$
MORAVEC	$0.004\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.002\frac{s}{ft}$	$0.003\frac{s}{ft}$	$0.004\frac{s}{ft}$

	NCC	SAD	SSD	NSSD	CENSUS	bester Extraktor
HARRIS	860	**1015**	980	660	115	**3630**
KLT	645	905	825	585	120	3080
MORAVEC	400	580	665	430	215	2290
bester Matcher:	1905	**2500**	2470	1675	450	9000

Tabelle 4.20: Datensatz (II): Tracking Ergebnisse für eine 360° Rotation um die y-Achse über 72 Frames in 10° Schritten: ohne Rematching, mit Bildpyramide, mit Re-Initialisierung der Punktmenge nach 2 Frames. Die Werte repräsentieren den Mittelwert und die Standardabweichung in $^\circ$ bzgl. des Idealwertes von 10°, sowie Mittelwerte für die Startpunktmenge und die durchschnittliche Anzahl an gelöschten inkonsistenten Features.

Mittelwert und Standardabweichung in Grad vom idealen Wert: 10°

Extraktor	NCC	SAD	SSD	NSSD	CENSUS
HARRIS	9.6893; 0.5421	9.6470; 0.4504	9.6875; 0.4723	9.6492; 0.5251	9.7093; 1.9228
KLT	9.7713; 0.5040	9.6859; 0.4842	9.7084; 0.5091	9.6515; 0.5966	9.8769; 2.1005
MORAVEC	9.8036; 0.5467	9.8856; 0.5064	9.7501; 0.5078	9.8071; 0.6389	6.4528; 312.5?

Prozentuale Angabe der gelöschten Features im Mittel bezogen auf die Startpunktmenge

HARRIS	1.57%	0.64%	1.02%	0.70%	8.18%
KLT	1.94%	0.66%	0.89%	0.76%	7.75%
MORAVEC	0.77%	0.71%	0.85%	0.84%	6.62%

Rechenzeit in Sekunden (s) pro Feature(ft)

HARRIS	$0.009\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.008\frac{s}{ft}$	$0.016\frac{s}{ft}$
KLT	$0.009\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.008\frac{s}{ft}$	$0.015\frac{s}{ft}$
MORAVEC	$0.008\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.005\frac{s}{ft}$	$0.007\frac{s}{ft}$	$0.013\frac{s}{ft}$

	NCC	SAD	SSD	NSSD	CENSUS	bester Extraktor
HARRIS	455	**975**	890	555	235	**3110**
KLT	710	940	735	455	270	**3110**
MORAVEC	530	870	740	515	125	2780
bester Matcher:	1695	**2785**	2365	1525	630	9000

Tabelle 4.22: Datensatz (II): Tracking Ergebnisse für eine 360° Rotation um die y-Achse über 72 Frames in 10° Schritten: mit Rematching, mit Bildpyramide, mit Re-Initialisierung der Punktmenge nach 2 Frames. Die Werte repräsentieren den Mittelwert und die Standardabweichung in $^\circ$ bzgl. des Idealwertes von 5°, sowie Mittelwerte für die Startpunktmenge und die durchschnittliche Anzahl an gelöschten inkonsistenten Features.

SIFT - Operator

Rotationsschritte	MW und STD	gelöschte Feature in %	Rechenzeit
5°	4.9881; 0.2609	2.17%	703.2s
10°	10.0056; 0.3214	2.15%	507.1s
15°	15.0441; 0.8483	2.14%	431.0s
20°	19.6488; 0.6585	2.50%	389.4s
25°	24.7127; 1.1019	5.08%	351.1s
30°	30.4570; 4.0043	3.02%	336.2s
35°	35.4116; 2.5547	1.62%	308.6s
40°	39.9159; 1.0803	1.87%	308.0s
45°	44.8489; 0.9941	1.37%	292.1s

Tabelle 4.23: Datensatz (II): Tracking Ergebnisse für eine $360°$ Rotation um die y-Achse über 72 Frames für den SIFT-Operator. Die Re-Initialisierung erfolgt jeweils nach $1, 2, 3, 4, 5, 6, 7, 8$und 9 Frames. Stabile Features werden auch über große Distanzen wiedergefunden.

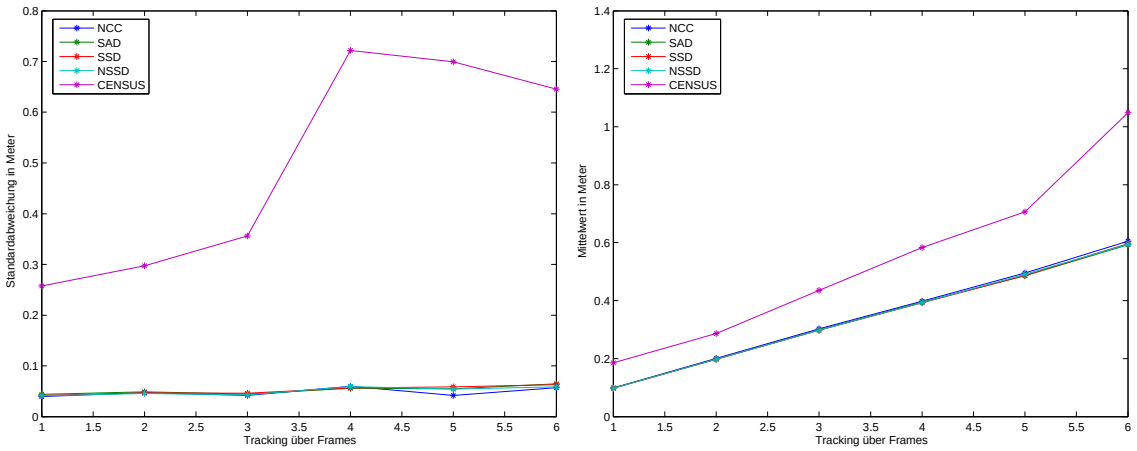
SIFT - Operator

Translationsschritte	MW und STD	gelöschte Feature in %	Rechenzeit
0.1m	0.0962; 0.0398	0.46%	322.3s
0.2m	0.1918; 0.0688	0.53%	231.6s
0.3m	0.3014; 0.0103	0.41%	201.4s
0.4m	0.4037; 0.0098	0.33%	181.8s
0.5m	0.4995; 0.0098	0.43%	172.3s
0.6m	0.6061; 0.0122	0.18%	154.0s
0.7m	0.7015; 0.0118	0.13%	148.2s
0.8m	0.8015; 0.0095	0.00%	133.2s
0.9m	0.8955; 0.0175	0.27%	144.0s

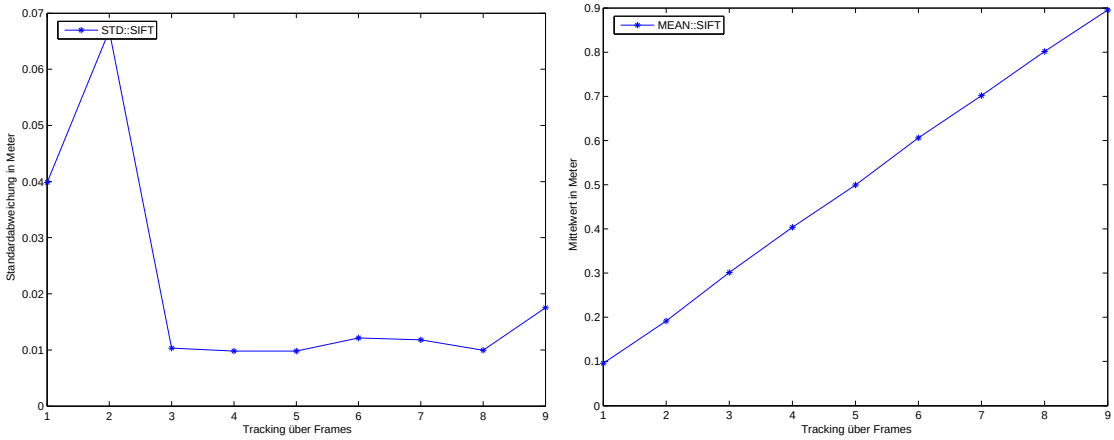
Tabelle 4.24: Datensatz (III): Tracking Ergebnisse für eine 3.6m Translation entlang der z-Achse über 36 Frames für den SIFT-Operator. Die Re-Initialisierung erfolgt jeweils nach $1, 2, 3, 4, 5, 6, 7, 8$und 9 Frames. Abbildung 4.3 zeigt die hier aufgeführten Werte in einem Plot.

Datensatz (III) Die Translationsergebnisse sprechen vor allem für den SIFT-Operator, der hier die Vorteile seiner Skalierungsinvarianz voll ausspielen kann. Er ist in der Lage die gemachte Bewegung bis zu $90cm$ relativ zur vorhergegangenen Position zu bestimmen, mit Abweichungen von unter $1cm$. In Tabelle 4.24 werden ausführlich die Ergebnisse der zurückgelegten Distanz aufgelistet. Für die anderen Kombinationen stehen zu diesem Zeitpunkt bereits keine Punkte mehr zur Verfügung, so dass eine Messung gar nicht erst möglich ist. Als vergleichbare Referenz zeigt Abbildung 4.3 die erzielten Ergebnisse aus KLT mit allen Matching-Metriken für die Standardabweichung und den Mittelwert bis zu einem Tracking über 6 Frames. Die simulierten Translationsschritte betragen in allen Fällen immer genau $10cm$

Auffallend sind auch hier wieder die instabilen Ergebnisse der Census-Transformation, die keine ausreichend genaue Positionsbestimmung des Systems erlauben. Alle anderen Metriken liefern vergleichbare und brauchbare Ergebnisse was deren Qualität betrifft. Dabei fällt besonders auf, dass die Abweichung sich immer um die $5cm$ bewegt. Die Rechenzeiten sind mit denen aus Datensatz (I) und (II) vergleichbar. Deswegen wird auf eine erneute detaillierte Auflistung verzichtet, da der Erkenntnisgewinn dadurch nicht gesteigert wird. Als einzige zusätzliche Erkenntnis lässt sich bereits jetzt feststellen, dass es sinnvoller ist die Re-Initialisierungsphase der Punktmenge Θ im Durchschnitt nach 4 bzw. 5 Frames durchzuführen. Die Schätzung liefert stabilere Ergebnisse, da klar zwischen den Bewegungen unterschieden werden kann. Außerdem läuft man beim Tracking nicht Gefahr in schwierigen Szenen keine stabile Punktmenge mehr bestimmen zu können.



(a)



(b)

Bild 4.3: Datensatz (III): (a) Kombination von KLT mit allen Matching-Metriken und Multiresolutionansatz, (b) SIFT-Tracking. Angegeben wird jeweils die Standardabweichung und der Mittelwert relativ geschätzt zur vorhergegangenen Lage- und Position über die angegebene Frameanzahl. Die vorgegebene Bewegung entspricht einer Translation über 36 Frames in Schritten à $10cm$, also für eine Gesamtdistanz von $3.6m$.

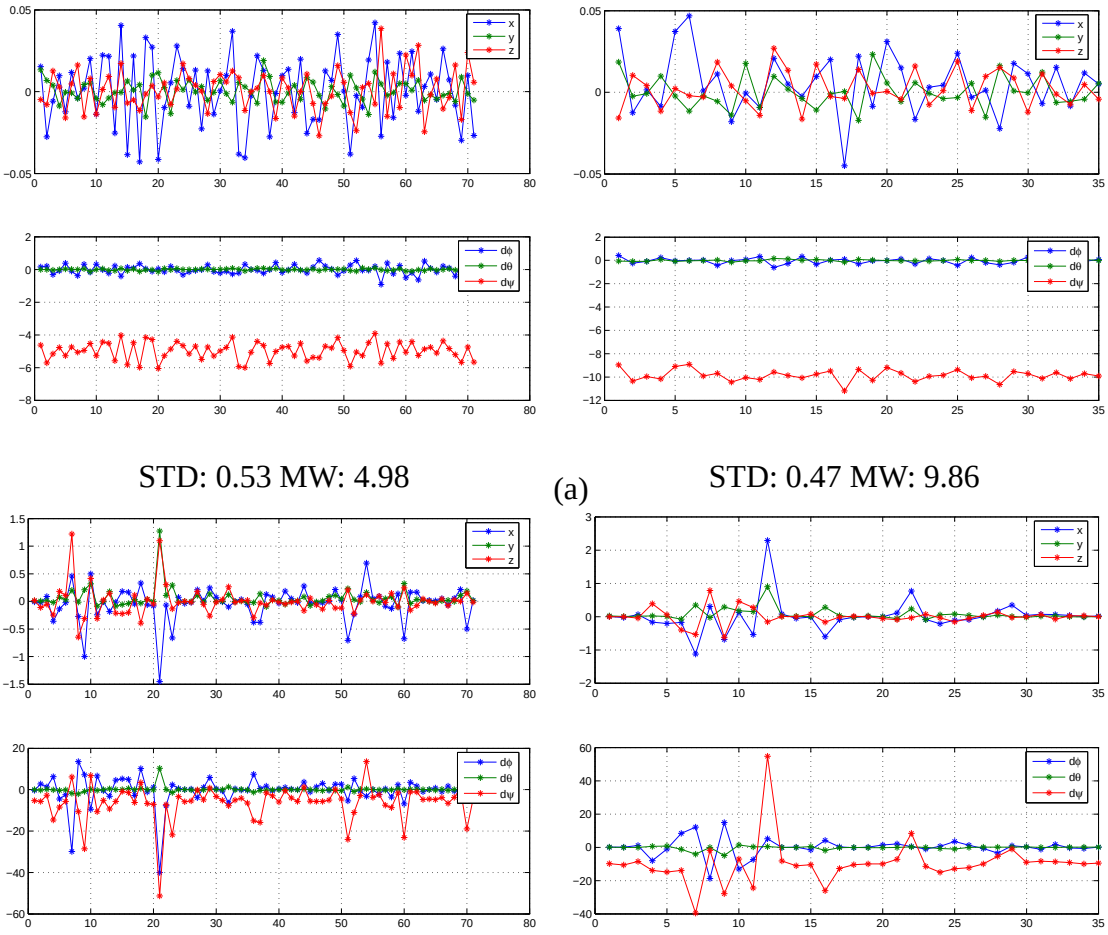
4.4 Validierung

Die im vorangegangen Abschnitt präsentierten Ergebnisse können an Hand der bekannten und vorher fest definierten Bewegung verifiziert werden. So lässt sich mit Hilfe der Plots erkennen, wie die Schätzung von Frame zu Frame verläuft. Exemplarisch werden hier Plots in Abbildung 4.4 der Census-Transformation und der Absolutsumme (SAD) in Kombination mit KLT betrachtet, da die Ergebnisse, wie aus den Tabellen zu erkennen, in den meisten Situationen relativ ähnlich ausfallen. Abbildung 4.5 zeigt für den gleichen Datensatz die geschätzte Position aus erzielten Korrespondenzen mit dem SIFT-Operator. Auf der im Anhang befindlichen CD-ROM sind alle weiteren Ergebnisse mit Plots protokolliert und können bei weiterem Interesse nachgesehen werden.

Auffallend sind gelegentliche extreme Ausreißer, der Census-Transformation nach oben und unten, so dass keine Aussage bezüglich der gemachten Bewegung relativ zur vorherigen Position möglich ist. Es lässt sich nicht genau sagen, wodurch diese Schwächen entstehen. Da die Punktmenge im Normalfall als ausreichend stabil betrachtet werden kann. Die Ausreißer sind generell immer mal wieder zu beobachten, was auch die teilweise hohen Standardabweichungen und Mittelwerte für die Census-Transformation erklärt. Bei genauerer Analyse zeigt sich jedoch, dass auch die Census-Transformation in der Lage ist ausreichend korrekte Punktkorrespondenzen zu liefern. So liefert eine Ergebnisfilterung mit 30% bezogen auf den idealen Wert, vergleichbar gute Lage- und Positionsschätzungen. Abbildung 4.6(c) zeigt eindeutig erkennbare Verbesserungen. Im Durchschnitt war die Schätzung in mindestens der Hälfte der Fälle möglich.

Ein weiterer Punkt, der auffällt ist, dass das Matching über mehrere Frames stabilere Ergebnisse liefert, als die Schätzung über eine geringe Frameanzahl, obwohl dieser mehr Punkte bei der Schätzung zur Verfügung stehen. Der Schätzalgorithmus ist in manchen Fällen nicht in der Lage zwischen Rotation und Translation zu unterscheiden, falls die Bewegung sehr klein ist.

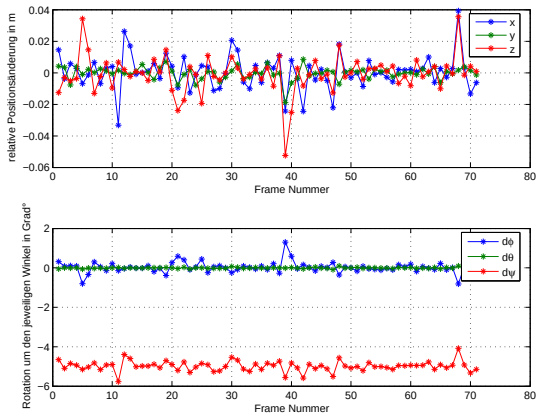
Die verifizierten Ergebnisse lassen eine Übertragung in die Realität als möglich erscheinen. Da die vorher definierten relativen Bewegungen mit geringen Abweichungen vom Idealwert erreicht werden. Dies gilt sowohl für verschieden Raumdimensionen, als auch für verschiedene Rotations -und Translationsbewegungen.



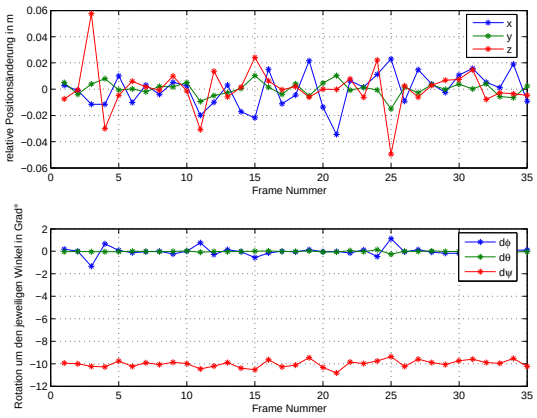
STD: 0.53 MW: 4.98 (a) STD: 0.47 MW: 9.86

STD: 8.62 MW: 6.35 (b) STD: 11.38 MW: 9.69

Bild 4.4: Datensatz (II): Kombination aus KLT+SAD (a), KLT+CENSUS (b) mit einem Multiresolutionansatz. Der obere Subplot gibt die Position in (x, y, z) und die vermutete Bewegung vom Ursprung in m. Die Werte für (a) sollten dabei alle 0 sein. Der untere Subplot gibt die Lage in Grad (ϕ, Θ, Ψ) relativ zur vorhergegangenen Position zwischen den einzelnen Frames an. (a) und (b) bewegen sich um $5°$ bzw. $10°$. Die Schwankungen ergeben sich aus zufälligen Fehlern und beeinflussen die Messung nicht.

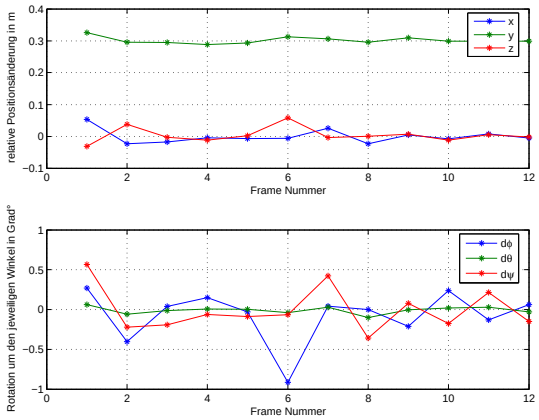


STD: 0.26 MW: 4.99

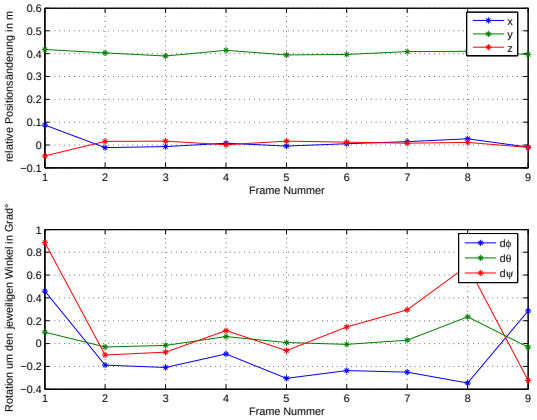


STD: 0.32 MW: 10.01

(a)



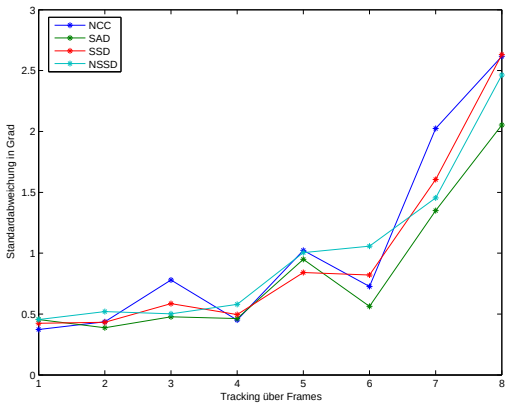
STD: 0.0103 MW: 0.3014



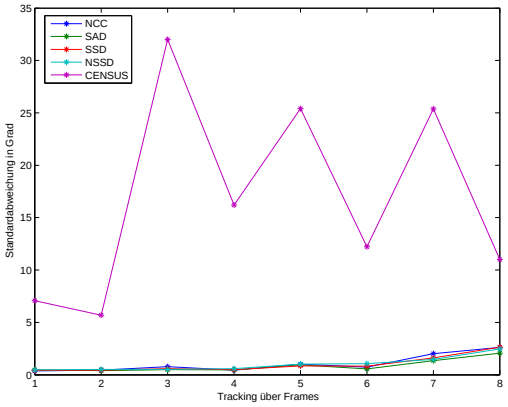
STD: 0.0098 MW: 0.4037

(b)

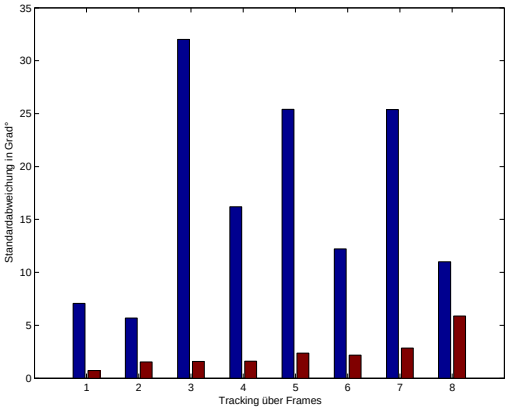
Bild 4.5: (a) Datensatz (II) - Ergebnisse SIFT: Tracking über 1 und 2 Frames für eine Rotation um $5°$ bzw. $10°$. (b) Datensatz (III) - Ergebnisse SIFT: Tracking über 3 und 4 Frames für eine Transaltion über $0.3m$ und $0.4m$. Der obere Subplot gibt die Position in (x, y, z) und die vermutete Bewegung vom Ursprung in m an. Die Werte für (a) sollten dabei alle 0 sein. Für (b) entsprechend bei 0.3 und 0.4. Der untere Subplot gibt die Lage in Grad (ϕ, Θ, Ψ) relativ zur vorhergegangenen Position zwischen den einzelnen Frames an. (a) bewegt sich um $5°$ bzw. $10°$. (b) sollte Werte um $0°$ liefern. Die Schwankungen ergeben sich aus zufälligen Fehlern und beeinflussen die Messung nicht.



(a)



(b)



(c)

STD (neu)	STD (alt)	Verworfen
0.738	5.073	35/72
1.542	9.739	13/36
1.577	15.49	12/24
1.603	20.01	7/18
2.377	22.95	9/14
2.184	28.92	4/11
2.857	32.26	5/10
5.872	37.20	3/8

(d)

Bild 4.6: Datensatz (II): Plot über mehrere Frames von KLT und Multiresolutionansatz in 5° Schritten: (a) Standardabweichung im Verhältnis zueinander ohne Census . (b) Standardabweichung im Verhältnis zueinander mit Census. (c) Standardabweichung von Census ungefiltert (blau) und gefiltert (rot) mit den Werten aus (d). Werte deren Standardabweichung mehr als 30% vom Idealwert abweichen, werden nicht betrachtet. Die Ausreißer erlauben in diesem Fall keine Schätzung der Position. Auffallend ist der ähnliche Anstieg der Standardabweichung wie bei den anderen Matching Verfahren aus (a). (d) gibt zusätzlich die Anzahl der verworfenen Schätzungen an. So werden bei den Schätzungen von Frame zu Frame 35 von 72 Messungen verworfen.

Kapitel 5

Zusammenfassung

Es konnte gezeigt werden, dass mit dem hier beschriebenen Verfahren des Intra- / Inter-Matching von Punktmengen eine schnelle und robuste Kombination aus verschiedenen Merkmalsextraktoren und Matchingverfahren möglich ist. Das Verfahren eignet sich somit zur relativen Bestimmung von Position und Lage eines Stereokamerasystems in Innenräumen. Da das Stereosystem als zusätzlich stützende Komponente in einem multisensoriellem Aufbau angesehen wird, sind die geschätzen Bewegungen als absolut ausreichend zu betrachten bei Standardabweichungen von $0.4°$ in kleineren Räumen bis hin zu Abweichungen von knapp einem $1°$ in großen Räumen. Hinzu kommen die unterschiedlichen Öffnugswinkel, die die genaue Positionsbestimmung vor allem im großen Raum erschwert haben, wodurch sich auch gleichzeitig der Einsatz von Weitwinkelobjektiven rechtfertigt, deren Kalibrierung zu einem weiteren Performanzgewinn führen wird.

Die Operatoren zur Extraktion der Punktmenge liefern eine gute Basis für deren Tracking in verschiedenen Bildfolgen. Unterschiede gibt es vor allem zwischen den beiden ähnlichen Operatoren Harris-Corner und KLT zum Moravec-Operator. Sie liefern in der Regel stabilere Merkmale, die sich eindeutiger durch Matching-Metriken für ein lokales Fenster beschreiben lassen. Wohingegen die Ergebnisse des Moravec oftmals zweideutig für das Matching sind und somit häufiger zu Fehlern bei der Korrespondenzbestimmung führen. Dies wirkt sich negativ auf die Schätzung aus. Einen verbesserten und erweiterten Ansatz verfolgt der SIFT-Operator, der Merkmale nicht nur über dessen lokale Nachbarschaft be-

schreibt, sondern mittels skalierungs- und rotationsinvarianter Transformationen diesen eindeutige Deskriptoren hinzufügt.

Für die Zuordnung von Punktmengen haben sich drei Matching-Metriken als brauchbar für eine schnelle Umsetzung erwiesen: SAD, SSD und die Census-Transformation. SAD und SSD eignen sich ebenfalls für die Schätzung der Position. Mit Einschränkungen eignet sich auch die Census-Transformation, die unter bestimmten Umständen Probleme aufweist und teils nicht nachvollziehbare Ausreißer erzeugt, so dass nur sehr instabile Schätzungen für die jeweilige Position gemacht werden können.

Jedoch vor dem Hintergrund, dass der Tracker in einer Hardwareplattform genutzt wird, bietet sich aufgrund der einfachen arithmetischen Operationen die Census-Transformation zusätzlich an. Die Stabilität des Operators wird wahrscheinlich durch ein größeres Korrelationsfenster weiter gesteigert, so dass es hier für weitere Optimierungen noch Spielraum gibt, was die Zuverlässigkeit angeht. Denkbar ist sicherlich auch eine Fusion aus Rank- und Census-Transformation, da sich beide über die Relation ihrer Nachbarpixel definieren und dies zu keiner zusätzlichen Berechnung bei der Beschreibung des Pattern führt.

Die Metriken NCC und NSSD liefern ebenfalls gute Ergebnisse, lassen sich aber in ihrer momentanen Implementierung, was ihr Laufzeitverhalten angeht, in keinem Echtzeitsystem einsetzen. Das Zuordnen von SIFT-Merkmalen erfolgt hingegen über den direkten Vergleich zweier Listen. Die eindeutigen Deskriptoren lassen eine sehr exakte Aussage darüber zu ob zwei Punkte miteinander korrespondieren. Besonders die erzielten Ergebnisse für die simulierte Translation sind fast ideal und übersteigen die Qualität aller anderen Kombinationen deutlich. Wohingegen die qualitativen Unterschiede bezogen auf die Rotation nicht so gravierend sind.

Für die Extraktion von Punktmengen kann man frei zwischen KLT, SIFT-Operator und Harris-Corner wählen. Als Empfehlung für korrekte Korrespondenzergebnisse bieten sich hier SAD und SSD an, sowie eine Option auf eine erweiterte Census-Transformation. Letztendlich bleiben als brauchbare Kombination für das Verfolgen von Punktmengen zur Lage- und Positionsbestimmung eines Stereokamerasystems im klassischen Sinne: KLT+SAD, KLT+SSD, Harris+SAD und Harris+SSD. Die etwas andere Idee des gegenseitigen Zuordnens zwischen Listen bestehend aus SIFT-Merkmalen liefert sehr gute Ergebnisse, die das System bei der Navigation gut stützen können, so dass der SIFT-Operator

ebenfalls empfohlen werden kann.

Die Basis für die Schnelligkeit zur Positionsbestimmung stellt dabei ein Multiresolutionansatz dar, der das Bild in verschiedene Auflösungstufen, vergleichbar mit einer Pyramide, einteilt. Es wird dabei die Annahme gemacht, dass markante Merkmale im Originalbild auch in niedrigen Auflösungsstufen zu finden sind und diese wieder den Merkmalen des Originalbildes zugeordnet werden können. Der Rechenaufwand des SIFT-Operators steigt bzw. fällt ebenfalls mit der Bildauflösung. Die hier erzielten Ergebnisse bestätigen vor allem seine Robustheit mit welcher Merkmale beschrieben und zugeordnet werden können. Approximationen dieses Verfahren sollten enorme Geschwindigkeitsverbesserungen erzielen, da die Rechenzeiten auf den Ausgangsbilddimensionen keine Echtzeitumsetzung erlauben. Der Versuch ebenfalls die Bildauflösung zu reduzieren hat in Kurztests jedoch die Genauigkeit der Schätzung negativ beeinflusst und Bedarf noch einer genaueren Untersuchung.

Für die weitere Entwicklung und das Management des Tracking stellt sich nun Frage, wie z. B. vermieden werden kann, dass in unstrukturierten Szenen die Punktmenge extrem gering ist und somit eine Schätzung erschwert wird. Eine mögliche Idee wäre sicherlich, die Extraktion für die Szene erneut zu starten, sobald diese unter einer bestimmten Schwelle liegen. Dafür müsste man nur die Kachelgröße reduzieren, so dass automatisch mehr Merkmale möglich sind. Gänzlich unstrukturierte Szenen stellen das Verfahren jedoch vor ein Problem. In dieser Situation entfällt das Stereosystem als stützender Sensor und liefert erst wieder Informationen in ausreichend strukturierten Szenen.

Nachdem der Raum einmal abgefahren und analysiert wurde, ist es nun von entschiedenem Interesse wie es gelingen kann, dass man die gemeinsamen Weltpunkte in einer Art Punktdatenbank speichert. Dabei sollten sie jederzeit wiederauffindbar sein. Sie ließen sich somit als stetige Referenz im Raum nutzen und als bekannte Stützpunkte verwenden. Diese Stützpunke müssten entsprechend kartografiert und beschrieben werden. Eventuell böte sich für diesen Fall nun der SIFT-Operator mit seinen eindeutigen Deskriptoren an. Durch das Arbeiten auf einer bekannten Größe von Merkmalen ließe sich die Rechenzeit vermutlich ausreichend reduzieren und mit Hilfe seiner Invarianzeigenschaften stabil lokalisieren.

Abschließend bleibt noch die Frage zu klären, wie sich ein solches Sensorsystem in nicht

geschlossenen Räumen verhalten könnte. Die erzielten Ergebnisse aus Datensatz (I) lassen vermuten, dass dies zu Problemen führen wird. Da es sich hier bereits um einen sehr großen Raum handelt, dessen Tiefe nur noch schwer zu bestimmen ist. Ebenso stellt das Punktmanagement ein Problem dar, da Außenszenzen eine wesentlich höhere Dynamik (z. B. Blätter im Wind) aufweisen als überwiegend statische Innenräume (z. B. Büroräume).

Anhang A

Verschiedenes

Grafische Oberfläche mit QT4

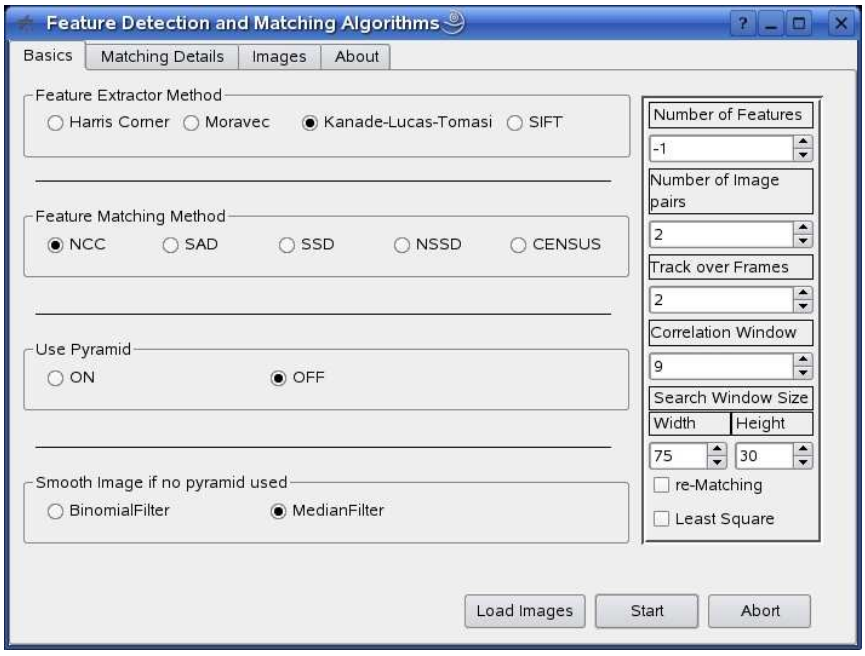


Bild A.1: Benutzeroberfläche ermöglicht Parameterveränderung

UML - Diagramm

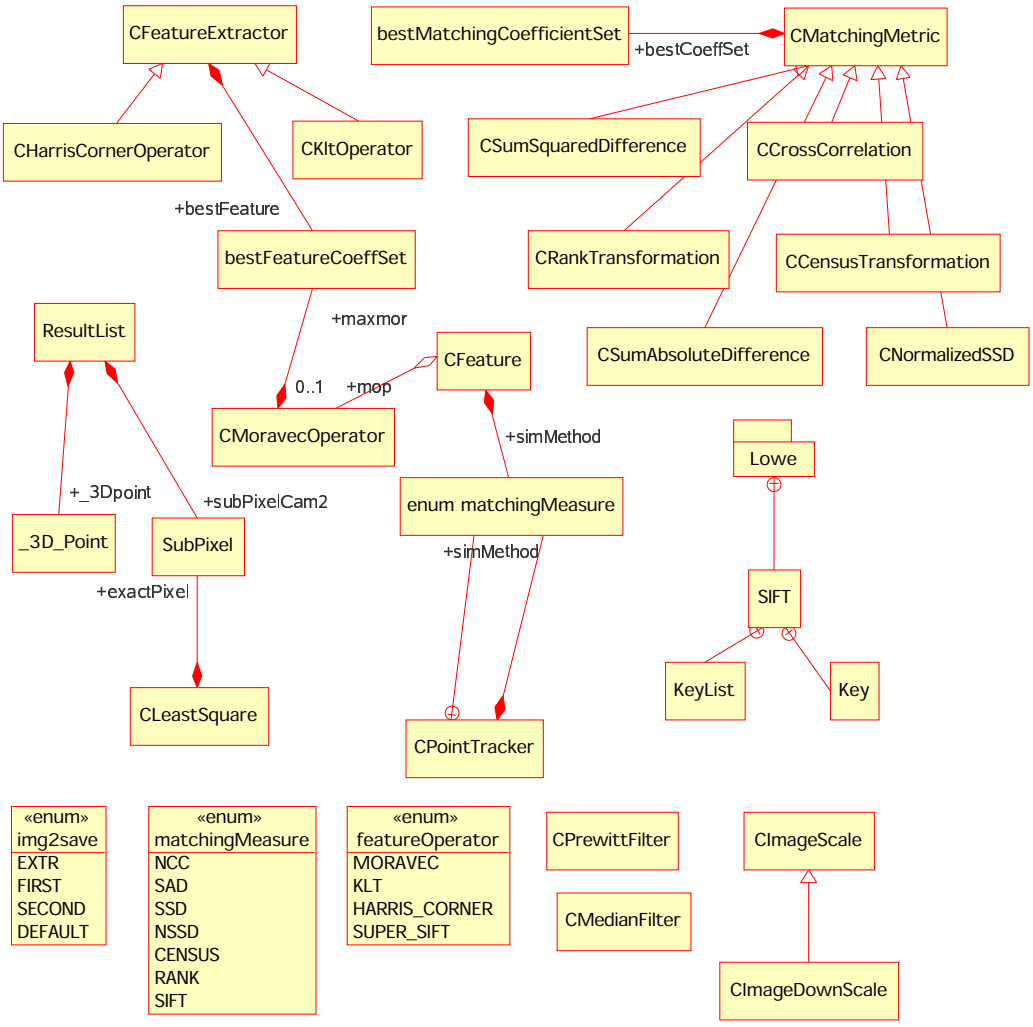


Bild A.2: UML - Klassendiagramm

Ergebnis - Ausgabeformat für .bin - Dateien

Header length: 136 byte
Data length : $n \cdot 10 \cdot 4$ byte

data item	**units**	**type**
Sensor size (x, y)	pixel	2 × double
Pixel size	m	double
Focal length	m	double
Lever, camera 1 (x, y, z)	m	3 × double
Lever, camera 2 (x, y, z)	m	3 × double
Pose, camera 1 (ω, ϕ, κ)	rad	3 × double
Pose, camera 2 (ω, ϕ, κ)	rad	3 × double
number of object points (n)	-	double
object point 1, location1, camera 1 (x, y)	pixel	2 × float
object point 1, location1, camera 2 (x, y)	pixel	2 × float
object point 1, location1, correlation	-	float
object point 1, location2, camera 1 (x, y)	pixel	2 × float
object point 1, location2, correlation	-	float
object point 1, location2, camera 2 (x, y)	pixel	2 × float
object point 1, location2, correlation	-	float
⋮		
object point n, location1, camera 1 (x, y)	pixel	2 × float
object point n, location1, camera 2 (x, y)	pixel	2 × float
object point n, location2, correlation	-	float
object point n, location2, camera 1 (x, y)	pixel	2 × float
object point n, location2, correlation	-	float
object point n, location2, camera 2 (x, y)	pixel	2 × float
object point n, location2, correlation	-	float

Tabelle A.1: Speicherformat für korrespondierende Punkte in Stereobildpaaren.

Bilddatensätze

Datensatz	Bewegung
B3	Rotation um 3 Achsen
B4	Rotation um y-Achse
B7	Translation x-Achse
C1	Kalibriermuster

Tabelle A.2: Übersicht für reale Bilddatensätze.

Software

- Kile 1.8.1 als LaTeX– Editor
- img2eps zum Konvertieren von Rastergrafiken ins .eps-Format
- The Gimp 2.2 zum Holen von Screenshots
- Xfig zum Erstellen von Vektorgrafiken
- QT Designer 4.1.4 zum Erstellen der Benutzeroberfläche
- FreeImage 3.9.0 als Bildklasse
- KDevelop 3.3.1 als Entwicklungsumgebung unter Linux OpenSUSE 10.1
- Microsoft Visual Studio 2003 .NET als Entwicklungsumgebung unter Windows XP
- valgrind 3.1.1 zur Speicherlecküberprüfung
- Matlab zur Lageschätzung und Auswertung der Ergebnisse

Anhang B

Danksagung

Mein besonderer Dank gilt Prof. Dr. Dietrich Paulus für sein Engagement und seine Zustimmung für die externe Diplomarbeit. Weiter danke ich Johannes Pellenz für die gute und kontinuierliche Betreuung meiner Diplomarbeit während dieser Zeit.

Außerdem möchte ich mich beim Team für optische Informationsysteme des DLR in Berlin Adlershof bedanken, dass ich so gut und freundlich aufgenommen wurde. Ganz besonders danke ich Anko Börner und Denis Griesbach für die vielen Hilfestellungen und De-Bug Sessions, sowie das zur Verfügung stellen des Schätzalgorithmus. Weiterhin danke ich Karsten Scheibe für die synthetischen Testbildsequenzen. Allen anderen danke ich natürlich für das gemeinsame Frühstück und den interessanten Anekdoten aus dem Alltag der ehemaligen DDR.

Meinen Eltern danke ich von ganzen Herzen, dass sie mich immer unterstützt haben und mir stets geholfen haben meine Ziele zu erreichen.

Weiterhin danke ich auch der Bundesrepublik Deutschland für die finanzielle Unterstützung während meines Studiums durch das BAföG-Amt.

Verzeichnis der Bilder

1.1 Sensor - Messaufbau 6

2.1 Epipolargeometrie 11
2.2 Kalibriermuster 13
2.3 Stereobildpaar 14
2.4 Rektifizierte Geometrie 17
2.5 Beispiel: rektifiziertes Stereobildpaar 18
2.6 Was ist ein Feature? 22
2.7 SIFT: Skalenraum, lokales Extremum 24
2.8 Beispiel: Bilder Feature Extraktoren 26
2.9 Tsukuba Bild 28
2.10 Rank-Transformation 30
2.11 Beispiel: Ranktransformiertes Bild 31
2.12 Census-Transformation 32
2.13 Disparity-Space-Image 34
2.14 Intrinsische Kurven 35
2.15 Beispiel: Graph Cut 36

3.1 Intra-Inter Matchingverfahren 42
3.2 Beispiel: Pixelrauschen 43
3.3 Beispiel: reale Aufnahme, klassische Tracking, Census 44
3.4 Beispiel: Bildpyramide 45
3.5 Matching in der Bildpyramide 47
3.6 Beispiel: Fehlerhaftes Matching 49

4.1 synthetische Stereoszene 53
4.2 Beispiel: Reale Aufnahme, Multiresolution Matching, Census, SAD . . . 57
4.3 III: Tracking Ergebnisse KLT: Translation z-Achse 72
4.4 KLT+SAD, KLT+CENSUS: relative Lage- und Position y-Rotation . . . 74
4.5 SIFT: relative Lage- und Position y-Rotation 75
4.6 gefilterte Census-Transformation 76

A.1 Benutzeroberfläche 81
A.2 UML - Klassendiagramm 82

Verzeichnis der Tabellen

2.1 Kalibrierung: intrinsische Kameraparameter 16

2.2 KLT - Eigenschaften 22

3.1 Berechnung des Pyramidenlevel 46

4.1 Kameraparameter 52

4.2 Versuchsparameter, synthetischer Datensatz 54

4.3 Feature-Entwicklung I 55

4.4 Feature-Entwicklung II 56

4.6 I: Tracking Ergebnisse Pyramide: 1 Frame, y-Achse 60

4.8 I: Tracking Ergebnisse Pyramide: 1 Frame, y-Achse, Re-Match 61

4.10 I: Tracking Ergebnisse Pyramide: 2 Frames, y-Achse 62

4.12 I: Tracking Ergebnisse Pyramide: 2 Frames, y-Achse, Re-Match 63

4.13 I: Tracking Ergebnisse SIFT: y-Achse, Re-Match 64

4.14 I: Tracking Ergebnisse SIFT: y-Achse, Re-Match, 362×362 64

4.16 II: Tracking Ergebnisse Pyramide: 1 Frame. y-Achse 66

4.18 II: Tracking Ergebnisse Pyramide: 1 Frame. y-Achse. Re-Match 67

4.20 II: Tracking Ergebnisse Pyramide: 2 Frames, y-Achse 68

4.22 II: Tracking Ergebnisse Pyramide: 2 Frames, y-Achse, Re-Match 69
4.23 II: Tracking Ergebnisse SIFT: y-Achse, Re-Match 70
4.24 III: Tracking Ergebnisse SIFT: Translation z-Achse 70

A.1 Speicherformat für korrespondierende Punkte 83
A.2 reale Bilddatensätze 84

Index

Bildpyramide, 44

diffraktives optisches Element, 14
DOE, 14

Epipolargeometrie, 10
    Epipolarlinie, 10
    Epipole, 10

Feature Extraktoren, 19
    Harris - Corner Operator, 21
    KLT - Operator, 20
    Moravec - Operator, 19
    SIFT - Operator, 22
Fine-Tuning, 45

Inter-Matching, 40
Intra-Matching, 40

Kalibriermuster, 13
Kalibrierung, 13
Kameramodell, 10
Kamerasysteme, 6
Korrespondenzbestimmung, 27

Lochkameramodell, 10

Matching Metriken, 27
    Census-Transformation, 30
    Dynamic Programming, 33
    Graph Cuts, 34
    Intrinsische Kurven, 34
    Least Square Matching, 32
    Normalized Cross Correlation, 28
    Normalized sum squared difference, 29
    Rank-Transformation, 29
    Sum of absolute difference, 29
    Sum of squared difference, 29
Matching Strategien, 27
Multiresolution Tracking, 44

Punktmenge, 40
    Re-Initialisierung, 41

reales Stereosystem, 52
Rektifizierung, 17
Rematching, 48

Sensorsystem, 6
Stereokamerasystem, 5
Stereosehen, 9
synthetisches Stereosystem, 52

Trajektorien, 56

Literaturverzeichnis

[AG92] P. Aschwanden and W. Guggenbuhl. Experimental results from a comparative study on correlation-type registration algorithms. In Wolfgang Förstner and Stephan Ruwiedel, editors, *Robust Computer Vision*, pages 268–289. Herbert Wichmann Verlag, 1992.

[Bal06] Dirk Balthasar. *Drei neue Verfahren zum Matching und zur Klassifikation unter Echtzeitbedingungen*. PhD thesis, Universität Koblenz, Verlag Fölbach Koblenz, 2006.

[BBH03] Darius Burschka, Myron Z. Brown, and G. D. Hager. Advances in computational stereo. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 25(8):993–1008, 8 2003.

[BT98] Stan Birchfield and Carlo Tomasi. Depth discontinuities by pixel-to-pixel stereo. In *ICCV*, pages 1073–1080, 1998.

[BVZ01] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE*, 23(11):1222–1239, 2001.

[CHRM96] Ingemar J. Cox, Sunita L. Hingorani, Satish B. Rao, and Bruce M. Maggs. A maximum likelihood stereo algorithm. *Comput. Vis. Image Underst.*, 63(3):542–567, 5 1996.

[DF01] Frederic Devernay and Olivier D. Faugeras. Straight lines have to be straight. *Machine Vision and Applications*, 13(1):14–24, 2001.

[FTV97] A. Fusiello, E. Trucco, and A. Verri. Rectification with unconstrained stereo geometry. In *British Machine Vision Conference*, pages 400–409, 1997.

[GN01] J.M. Gluckman and S. Nayar. Rectifying transformations that minimize resampling effects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume I, pages 111–117, 12 2001.

[Gru85] Armin Gruen. Adaptive least squares correlation: A powerful image matching technique. *South Africa Journal of Photogrammetry, Remote Sensing and Cartography*, 14(3):175–187, 1985.

[Hir03] Heiko Hirschmüller. *Stereo Vision Based Mapping and Immediate Virtual Walkthroughs*. PhD thesis, School of Computing De Montfort University Leicester, 2003.

[HS88] C. Harris and M. Stephens. A combined corner and edge detector. In *Fourth Alvey Vision Conference*, pages 147–151, Manchester, UK, 1988.

[HZ03] Richard I. Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.

[KZ01] Vladimir Kolmogorov and Ramin Zabih. Computing visual correspondence with occlusions via graph cuts. In *ICCV*, pages 508–515, 2001.

[Low04] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[Mor80] Hans P. Moravec. *Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover*. PhD thesis, Carnegie Mellon University, Robotics Institute, Pittsburgh, PA, 5 1980. Available as Stanford AIM-340, CS-80-813 and CMU-RI-TR-3.

[MS04] Krystian Mikolajczyk and Cornelia Schmid. Scale & affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

[NBN06] David Nistér, James R. Bergen, and Oleg Naroditsky. Visual odometry for ground vehicle applications. *Journal of Field Robotics*, 23(1), 2006. inaugural issue.

[OK85] Y. Ohta and T. Kanade. Stereo by intra- and inter-scanline search using dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7:139–154, 1985.

[OPM02] Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. A generalized local binary pattern for multiresolution gray scale and rotation invariant texture classification. Technical report, Machine Vision and Median Processing Unit Infotech Oulu, University of Oulu, Finland, 2002.

[OPX00] Timo Ojala, Matti Pietikäinen, and Z. XU. Rotation-invariant texture classification using feature distribution. Technical report, Machine Vision and Median Processing Unit Infotech Oulu, University of Oulu, Finland, 2000.

[Ora01] D. Oram. Rectification for any epipolar geometry. In *British Machine Vision Conference*, page 7th Session: Geometry and Structure, 2001.

[Pau03] Dietrich Paulus. Structure from motion, 5 2003.

[RC98] Sébastien Roy and Ingemar J. Cox. A maximum-flow formulation of the n-camera stereo correspondence problem. In *ICCV*, pages 492–502, 1998.

[SB00] R. Schuster and B. Braunecker. Calibration of the lh systems ads40 airborne digital sensor. *International Society for Photogrammetry and Remote Sensing*, XXXIII:288–294, 2000. Amsterdam.

[SMB00] Cordelia Schmid, Roger Mohr, and Christian Bauckhage. Evaluation of interest point detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000.

[SSZ01] D. Scharstein, Richard Szeliski, and R. Zabih. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, 2001. In Proceedings

of the IEEE Workshop on Stereo and Multi-Baseline Vision, Kauai, HI, Dec. 2001.

[ST94] Jianbo Shi and Carlo Tomasi. Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94)*, pages 593–600, Seattle, 6 1994.

[STS05] B. Strackenbrock, B. Tsuchiya, and K. Scheibe. 3d-modelling and visualisation from 3d-laser scans and panoramic images. In U. Reulke, R.; Knauer, editor, *Panoramic Photogrammetry Workshop, Berlin, Germany, 24-25 February 2005*, volume XXXVI-5/W8, 2005. LIDO-Berichtsjahr=2005;.

[SZS03] Jian Sun, Nan-Ning Zheng, and Heung-Yeung Shum. Stereo matching using belief propagation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(7):787–800, 2003.

[TM98] Carlo Tomasi and Roberto Manduchi. Stereo matching as a nearest-neighbor problem. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(3):333–340, 1998.

[Tsa87] Roger Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323–344, 1987.

[TV98] E. Trucco and A. Verri. *Introductory Techniques for 3–D Computer Vision*. Prentice Hall, New York, 1998.

[WB01] Greg Welch and Gary Bishop. An introduction to the kalman filter. Technical report, University of North Carolina at Chapel Hill, 2001.

[ZDFL95] Zhengyou Zhang, Rachid Deriche, Olivier D. Faugeras, and Quang-Tuan Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78(1-2):87–119, 1995.

[Zha00] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[ZW00] Ramin Zabih and John Woodfill. A non-parametric approach to visual correspondence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.