Universität der Bundeswehr München

Automatische Kameraposeschätzung für komplexe Bildmengen

M. Sc. Mario Michelini

Vollständiger Abdruck der von der Fakultät für Informatik der Universität der Bundeswehr München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Gutachter:

1. Prof. Dr.-Ing. Helmut Mayer
2. Prof. Dr.-Ing. Reinhard Koch (Christian-Albrechts-Universität Kiel)

Die Dissertation wurde am 17.04.2018 bei der Universität der Bundeswehr München eingereicht und durch die Fakultät für Informatik am 05.07.2018 angenommen. Die mündliche Prüfung fand am 19.07.2018 statt.

Zusammenfassung

Diese Dissertation behandelt die Konzeption, Implementierung und Analyse eines hierarchischen Verfahrens für die automatische Kameraposeschätzung für komplexe Bildmengen. Unter letzteren sind Mengen von Bildern zu verstehen, die komplizierte Aufnahmekonfigurationen, aber auch eine Kombination von Bildern unterschiedlicher Kameras enthalten können. Mit Ausnahme der Kamerakalibrierung wird keine Zusatzinformation in Form von Daten von Global Positioning System (GPS) oder Inertial Navigation System (INS) sowie über die Aufnahmekonfiguration benutzt. Näherungswerte für die Kamerakalibrierung können oft auch aus den Metadaten der Bilder automatisch abgeleitet und im Rahmen der Kameraposeschätzung weiter verfeinert werden.

Neben einer hohen Genauigkeit für die geschätzten Kameraposen liegt der Fokus dieser Forschungsarbeit auf der effizienten Bestimmung einer möglichst vollständigen Bildverknüpfung selbst beim Vorliegen von instabilen Aufnahmekonfigurationen oder großen (geometrischen bzw. radiometrischen) Bildverzerrungen. Instabile Aufnahmekonfigurationen sind charakterisiert durch einen unzureichenden Abstand zwischen den Aufnahmen und werden über die Klassifikation mit einem Random Forest detektiert. Die Reduktion des Aufwandes zur Bestimmung der paarweisen Beziehungen erfolgt auf Grundlage der Bildähnlichkeiten. Letztere werden durch die Einbettung der Merkmalsdeskriptoren in den Hamming-Raum effizient ermittelt.

Um eine höhere Genauigkeit und Robustheit zu erzielen, basiert die Kameraposeschätzung auf Scale-Invariant Feature Transform (SIFT) sowie dreifachen Bildverknüpfungen in Form von Triplets. Die Modellierung der Verknüpfungen zwischen den Bildern erfolgt hierzu über einen gewichteten ungerichteten Graphen. Ein iterativer Ansatz formuliert die Ermittlung von geeigneten Bildverknüpfungen als Suche nach einem terminalen minimalen Steinerbaum im Kantengraphen. Während minimale Spannbäume hierzu bereits häufiger eingesetzt wurden, stellt dies die erste praktische Anwendung von Steinerbäumen im Kontext der Kameraposeschätzung dar.

Vorhandene Bildschleifen werden über Graphenanalyse detektiert und über zusätzliche Bildverknüpfungen geschlossen. Durch die Schleifenschlüsse erfolgt eine Erhöhung der Genauigkeit der geschätzten Kameraposen. Zuletzt wird die Laufzeit der hierarchischen Vereinigung zur Kameraposenschätzung über die Optimierung der Vereinigungsreihenfolge von Bildern verbessert. Die dazu eingesetzte Suche nach maximaler Paarung im Graphen führt zu Vereinigungen, die einen höheren Parallelisierungsgrad ermöglichen.

Das Potential des entwickelten Ansatzes wird anhand von praktischen Experimenten dargestellt. Diese sind auch die Grundlage für eine Analyse der vorgestellten Konzepte. Die Einordnung in den Stand der Forschung erfolgt über den Vergleich mit einigen aktuellen Verfahren für die automatische Kameraposeschätzung.

Abstract

This thesis deals with the design, implementation and analysis of a hierarchical approach for automatic pose estimation (structure from motion) for complex image sets. The latter means sets of images that can contain complex view configurations, but also a combination of images from different cameras. With the exception of camera calibration, no additional information in the form of data from Global Positioning System (GPS) or Inertial Navigation System (INS) as well as about the view configuration is used. Approximate calibration parameters can often be derived automatically from the metadata of the images and further refined during pose estimation.

In addition to high accuracy for the estimated camera poses, the focus of this thesis lies on the efficient determination of the complete connectedness of the images even if instable view configurations or large (geometric or radiometric) image distortions exist. Instable view configurations are characterized by an insufficient distance between views and are detected by classification with a random forest. The reduction of the effort for the determination of the pairwise relations is based on image similarities. The latter are efficiently estimated by embedding the image descriptors into Hamming space.

To obtain a higher accuracy and robusteness, pose estimation is based on the Scale-Invariant Feature Transform (SIFT) as well as threefold image connections in the form of triplets. The modeling of the connections between images employs a weighted undirected graph. An iterative approach formulates the determination of suitable image connections as search for a terminal minimum Steiner tree in the line graph. While minimum spanning tress have already been used frequently, this is the first practical application of Steiner trees in the context of camera pose estimation.

Existing image loops are detected using graph analysis and closed by additional image connections. Loop closing increases the accuracy of the estimated camera poses. Finally, the runtime of the hierarchical merging for pose estimation is improved by optimizing the merging order of the images. The employed search for maximum matching in the graph leads to mergings allowing for a higher degree of parallelization.

The potential of the developed approach is demonstrated by means of practical experiments. These are also the basis for an analysis of the presented concepts. The approach is related to the state-of-the-art by comparing it with several current approaches for automatic pose estimation.

Inhaltsverzeichnis

1 Einleitung

Die Ermittlung von 3D-Informationen für Gebäuden und Sehenswürdigkeiten oder von Höheninformationen aus Bildern ist für Planungen und Visualisierungen von zunehmender Bedeutung. Aufgrund der niedrigen Kosten für digitale Kameras und dem geringen technischen Aufwand für die Aufnahme stellen Bilder eine immer häufiger eingesetzte Datenquelle dar. Die Entstehung von Internetplattformen zum Austausch von Bildern wie Flickr[1] führte zudem zum rasanten Wachstum an öffentlich verfügbaren Bildmengen. Die enthaltenen Bilder können jedoch von verschiedenen Quellen stammen und verfügen teilweise über keine oder ungenaue Zusatzinformation. Solche Bildmengen werden in dieser Arbeit als *komplexe Bildmengen* bezeichnet, da beliebige Aufnahmekonfigurationen von verschiedenen Aufnahmegeräten und zu unterschiedlichen Zeitpunkten vorkommen können.

Die Informationsgewinnung basiert auf der Schätzung von Kameraposen, die über die 3D-Rekonstruktion auch eine spärliche 3D-Punktwolke liefert. Dieser Prozess wird im Rahmen dieser Forschungsarbeit *Kameraposeschätzung* genannt. Abb. 1.1 stellt eine rekonstruierte Punktwolke für eine komplexe Bildmenge zusammen mit den geschätzten Kameraposen dar. Ursprünge der Kameraposeschätzung liegen in der Photogrammetrie, wo die Schätzung lange Zeit auf Grundlage manuell gemessener Verknüpfungsinformation erfolgte. Später konnte diese für Luftbilder durch die Nutzung von automatisch zugeordneten Bildpunkten, Navigationsinformation von Global Positioning System (GPS), Inertial Navigation System (INS) sowie im Nahbereich durch kodierte Marken weitgehend automatisiert werden. (Fitzgibbon und Zisserman 1998) und (Koch et al. 1998) gehörten zu den frühen Arbeiten, die unter Annahme von Sequenzen von Bildern aufgenommen mit handelsüblichen Kameras eine automatische Kameraposeschätzung realisierten. Schaffalitzky und Zisserman (2002) zeigten dann erstmals, dass automatische Kameraposeschätzung auch ohne Verwendung jeglicher Zusatzinformation möglich ist.

Während in der klassischen Photogrammetrie hoch genaue Kameraposeschätzung angestrebt wird, sucht man heutzutage vermehrt nach Möglichkeiten, große Bildsammlungen aus dem Internet zu visualisieren sowie nach bestimmten Bildern oder Szenebereichen suchen zu können. Die Grundlage hierzu stellen wiederum zum Teil die Kameraposen dar. Insbesondere entstand der Bedarf an automatischen Verfahren, die mit derart großen Datenmengen möglichst effizient umgehen können. Den Grundstein legten Snavely et al. (2006) mit dem System *Photo Tourism*, das sehr genaue Kameraposen lieferte, jedoch für große Bildmengen unzureichend skalierte. Es folgten Arbeiten mit dem Ziel, immer größere Bildsammlungen unter möglichst geringen Qualitätsverlusten effizient verarbeiten zu können (siehe Kapitel 3).

Kameraposeschätzung aus Bildern stellt für viele Anwendungen außerdem eine günstige Alternative im Vergleich zu teuren Laserscannern dar. Letztere sind in der Lage, die 3D-Punktwolke direkt

[1] www.flickr.com

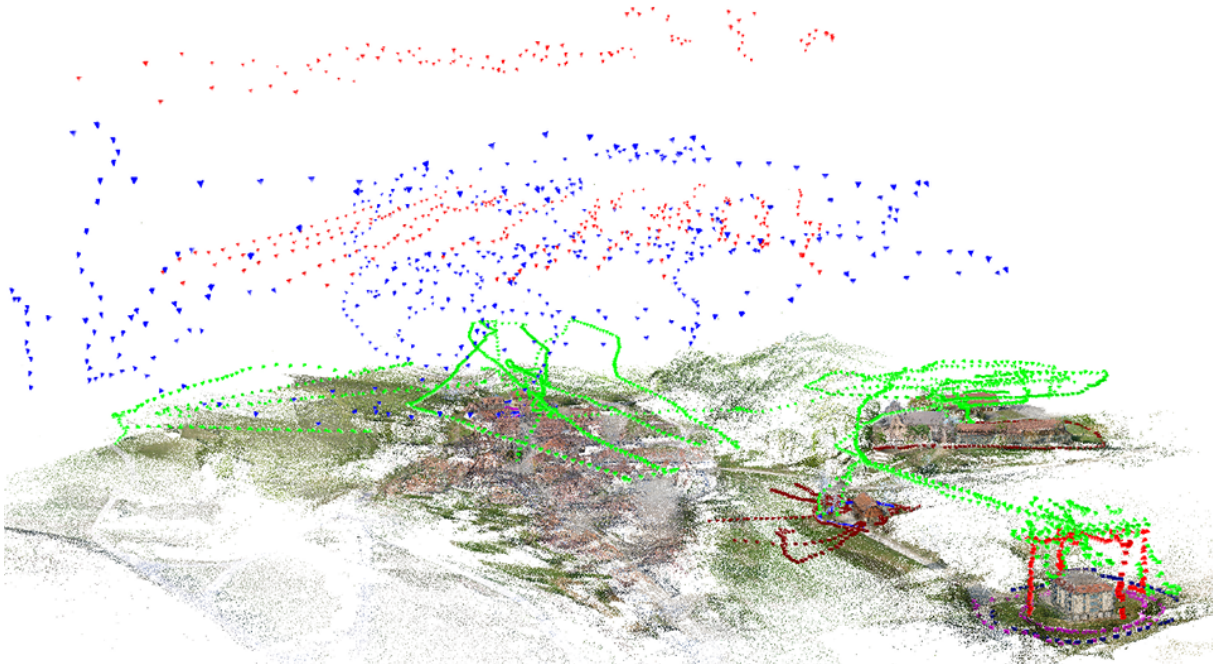


Abbildung 1.1: Kameraposen und rekonstruierte 3D-Punktwolke für eine komplexe Bildmenge. Kameras sind durch farbige Pyramiden dargestellt, wobei jede Farbe für einen anderen Kameratyp steht. Die Kameraposeschätzung erfolgte mit dem im Rahmen dieser Arbeit entwickelten Verfahren.

über die Abstandsmessung mit dem Laser zu erstellen. Dies erfordert jedoch hohe Investitionen in Geräte und Expertenwissen. Für die Kameraposeschätzung sind hingegen Aufnahmen von handelsüblichen Kameras ausreichend, die auch von unerfahrenen Benutzern stammen können. Dies kann auch für die 3D-Rekonstruktion von topographischen Objekten, wie beispielsweise Gebäuden, genutzt werden, wenn statt einer Luftbildkamera, montiert auf einem Flugzeug, eine Drohne (engl. *unmanned aerial vehicle – UAV*) mit einer deutlich günstigeren Kamera benutzt wird. Während für das Laserscanning spezielle Hardware und Expertenwissen erforderlich sind, benötigt man für die 3D-Rekonstruktion aus Bildern leistungsfähige Rechner. Letztendlich bietet die automatische Kameraposeschätzung für eine Reihe von Anwendungen eine kostengünstige Alternative mit der Ergebnisse von vergleichbarer Qualität erzielbar sind.

1.1 Problemstellung

Ein Verfahren für die automatische Kameraposeschätzung in komplexen Bildmengen muss verschiedene Herausforderungen meistern. Für praktische Anwendungen ist eine euklidische Rekonstruktion notwendig. Somit müssen zuerst intrinsische Kameraparameter bestimmt bzw. verfeinert werden, wenn diese nicht oder nur ungenau vorliegen. Ersteres erfolgt bei der Poseschätzung mittels sogenannter Autokalibrierung, die jedoch eine ausreichende 3D-Struktur der abgebildeten Szene voraussetzt (siehe Abschnitt 2.1.1). Da dies für allgemeine Szenen nicht garantiert werden kann, wird in der vorliegenden Arbeit von gegebenen Näherungswerten für

die intrinsischen Kameraparameter (z. B. aus den Metadaten der Bilder) ausgegangen, die unter Umständen im Rahmen der Kameraposeschätzung verfeinert werden.

Weiterhin muss das Verfahren über eine robuste Bildzuordnung verfügen, um auch Bilder mit größeren relativen Verzerrungen verarbeiten zu können (siehe Abschnitt 2.1.2). Unter Letzteren werden hier relative Änderungen zwischen den Bildern verstanden, die durch große Abstände zwischen Aufnahmepositionen und die daraus resultierenden perspektivischen Verzerrungen verursacht werden. Zudem können Aufnahmen zu unterschiedlichen Zeitpunkten oder mit verschiedenen Kameras starke radiometrische Unterschiede verursachen und somit die Bildzuordnung zusätzlich erschweren. Das einzusetzende Bildzuordnungsverfahren hängt daher von der Art der zu verarbeitenden Daten ab. Handelt es sich beispielsweise um Videoaufnahmen, so können einfache Verfahren eingesetzt werden, da keine größeren Bildverzerrungen zu erwarten sind. Im Falle von Einzelbildern steigt die Komplexität des nötigen Verfahrens mit dem Grad an gewünschter zulässiger Toleranz der Verzerrung zwischen den Bildern zum Teil stark an. Bei komplexen Bildmengen ist von Einzelbildern unterschiedlicher Kameras auszugehen, die zudem von sehr unterschiedlichen Aufnahmenzeitpunkten stammen können. Daher ist es für eine vollständige Kameraposeschätzung essenziell ein robustes Bildzuordnungsverfahren zu benutzen.

Eine grundlegende Annahme dieser Forschungsarbeit besteht darin, dass aufgrund unbekannter Aufnahmekonfiguration kein Wissen über die Überlappungen zwischen den Bildern existiert. Die Bestimmung der fehlenden Bildverknüpfungen bestimmt daher zusammen mit dem eingesetzten Bildzuordnungsverfahren maßgeblich die Effizienz der Kameraposeschätzung. Der Einfluss ist umso stärker, je robuster und somit rechenaufwändiger das eingesetzte Bildzuordnungsverfahren ist. Aus diesem Grund ist es für eine effiziente automatische Schätzung von Kameraposen notwendig, einen Ansatz zu entwickeln, der mit möglichst wenigen aufwändigen Bildzuordnungen eine zuverlässige Poseschätzung ermöglicht.

Komplexe Bildmengen können Aufnahmekonfigurationen enthalten, die einen negativen Einfluss auf die Kameraposeschätzung haben. Diese sogenannten kritischen Kamerakonfigurationen (vgl. Abschnitt 2.1.6) weisen einen zu geringen Abstand zwischen den Aufnahmepositionen im Vergleich zum Abstand des aufgenommenen Szeneobjekts auf. Dadurch können sie zu signifikanten Abweichungen der geschätzten Kameraposen bis hin zum Scheitern der Poseschätzung führen. Aus diesem Grund muss das Verfahren in der Lage sein, solche Konfigurationen zuverlässig zu detektieren und herauszufiltern.

Zusammenfassend geht die vorliegende Arbeit von komplexen Bildmengen mittlerer Größe (d. h. einige tausend Bilder) aus, die sich aus einer Kombination von Luft-, Drohnen- und Bodenaufnahmen (vgl. Abschnitt 6.1) zusammensetzen können sowie eine geringe Redundanz aufweisen. Letzteres bedeutet, dass die meisten Bilder essenziell sind und daher eine vollständige Bildverknüpfung anzustreben ist. Durch die Kombination von Aufnahmen aus der Luft mit Bodenaufnahmen sind außerdem große Bildverzerrungen zu erwarten, die eine entsprechend robuste Bildzuordnung erfordern. Weiterhin wird von keiner vordefinierten Aufnahmekonfiguration ausgegangen, was die Detektion kritischer Kamerakonfigurationen erforderlich macht. Mit Ausnahme von Näherungswerten für die Kamerakonstante wird keine Zusatzinformation verwendet. Basierend darauf besteht das Ziel der Forschungsarbeit in der effizienten automatischen Bestimmung von Bildverknüpfungen für eine robuste und zuverlässige Kameraposeschätzung.

1.2 Beiträge der Arbeit

Um Kameraposen auch im Falle von großen Bildverzerrungen zuverlässig schätzen zu können, wird das robuste Verfahren von Mayer et al. (2012) benutzt. Es basiert auf dreifachen Bildverknüpfungen in Form von Triplets und ist in der Lage, Korrespondenzen selbst bei größeren perspektivischen oder radiometrischen Bildunterschieden zuverlässig zu bestimmen (vgl. Abschnitt 2.1.2). Die Vereinigung von Bildern erfolgt anhand des hierarchischen Ansatzes von Mayer (2014), der aufgrund der Parallelisierbarkeit eine effiziente Kameraposeschätzung ermöglicht (vgl. Abschnitt 2.1.8). Somit stellt diese Forschungsarbeit eine Erweiterung der Verfahren von (Mayer et al. 2012) und (Mayer 2014) mit folgenden neuen Beiträgen dar:

- Automatische Bestimmung von Bildverknüpfungen inklusive Schleifenschluss
- Effiziente Reduktion der Zahl der notwendigen geometrischen Verifikationen
- Detektion kritischer Kamerakonfigurationen
- Verbesserung der Lastverteilung und der Robustheit der hierarchischen Vereinigung

In (Mayer et al. 2012) wird von existierenden Bildverknüpfungen ausgegangen, die über den im Rahmen dieser Dissertation entwickelten Ansatz ermittelt werden. Die Bestimmung optimaler Verknüpfungen zwischen den Bildern für eine zuverlässige Kameraposeschätzung wurde als Suche nach dem terminalen minimalen Steinerbaum in einem ungerichteten gewichteten Graphen formuliert. Während minimale Spannbäume hierzu bereits häufiger eingesetzt wurden (vgl. Abschnitt 3.3), stellt dies die erste Anwendung von Steinerbäumen im Zusammenhang mit einer Kameraposeschätzung dar.

Die robuste Bildzuordnung von Mayer et al. (2012) basiert auf komplexen, zeitaufwändigen Rechenoperationen, um mit großen Bildverzerrungen umgehen zu können. Daher ist es für die Effizienz der Kameraposeschätzung von großer Bedeutung, die Anzahl notwendiger geometrischer Verifikationen mittels der aufwändigen robusten Bildzuordnung gering zu halten. Dazu werden Ähnlichkeiten zwischen Bildern geschätzt und basierend auf diesen wird entschieden, wo eine geometrische Verifikation erforderlich und sinnvoll ist. Die Ähnlichkeitsschätzung basiert auf den Merkmalskorrespondenzen, ermittelt über den Vergleich von binären Merkmalsdeskriptoren. Zu diesem Zweck wurde eine Vorgehensweise zur Quantisierung von reellen auf binäre Deskriptoren entwickelt, die auf Grundlage von Orthanten in mehrdimensionalen Deskriptorräumen eine szeneunabhängige Einbettung ermöglicht.

Die Detektion kritischer Kamerakonfigurationen wurde über die Klassifikation mittels Random Forest (vgl. Abschnitt 2.5.3) umgesetzt. Hierbei handelt es sich um eine neuartige Vorgehensweise, die im Vergleich zur schwellenwertbasierten Klassifikation (Michelini und Mayer 2016), eine zuverlässigere Erkennung von Paaren mit ungeeigneter Aufnahmekonfiguration ermöglicht. Zudem ist die angewendete Vorgehensweise effizienter als die Detektion basierend auf der Modellauswahl, die die Bestimmung mehrerer Modelle erfordert (vgl. Abschnitt 3.2.1).

Zuletzt erfolgte die Verbesserung der Lastverteilung während der hierarchischen Vereinigung (Mayer 2014) von Bildern. Diese basiert auf hierarchischer Clusteranalyse, die lediglich eine suboptimale Ausnutzung der zur Verfügung stehenden Ressourcen ermöglicht. Über die Formulierung der Verteilung der Vereinigungsoperationen als Paarung im ungerichteten gewichteten Graphen

konnte neben einem höheren Parallelisierungsgrad auch eine Verbesserung der Robustheit erzielt werden.

Teile dieser Forschungsarbeit wurden bereits veröffentlicht in:

> M. Michelini und H. Mayer (2016). "Efficient Wide Baseline Structure from Motion". In: *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* III-3, S. 99–106
>
> H. Mayer und M. Michelini (2016). "Orientierung großer Bildverbände". In: *Handbuch der Geodäsie*. Hrsg. von W. Freeden und R. Rummel. Springer Berlin Heidelberg, S. 197–228
>
> M. Michelini und H. Mayer (2014). "Detection of Critical Camera Configurations for Structure from Motion". In: *ISPRS International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XL-3/W1, S. 73–78

1.3 Gliederung der Arbeit

Die Arbeit ist in sieben Kapitel unterteilt. Anschließend an die Einleitung befasst sich das zweite Kapitel mit den Grundlagen, die zum Verständnis dieser Dissertation erforderlich sind. Der aktuelle Stand der Forschung im Bereich des Forschungsgegenstandes ist das Thema des dritten Kapitels.

Der Kern der eigenen Arbeiten wird in den Kapiteln 4 bis 6 beschrieben. Das vierte Kapitel befasst sich mit der theoretischen Herleitung der Konzepte für die automatische Bestimmung von Bildverknüpfungen sowie der Schätzung der Bildähnlichkeiten. Die praxisrelevante Umsetzung wird im fünften Kapitel erläutert. Dieses Kapitel beschreibt zudem die Detektion kritischer Kamerakonfigurationen sowie den Ansatz zur Verbesserung der Lastverteilung während der hierarchischen Vereinigung von Bildern.

Das sechste Kapitel enthält die Evaluierung der entwickelten Ansätze und die Bewertung der damit erzielten Ergebnisse. Darüber hinaus findet in diesem Kapitel auch ein Vergleich mit einigen existierenden Verfahren zur Kameraposeschätzung statt, woraus sich die Einordnung in den aktuellen Stand der Technik ergibt. Im letzten Kapitel folgen abschließende Bemerkungen und ein Ausblick.

2 Grundlagen

In diesem Kapitel werden Grundlagen zum Verständnis der vorliegenden Dissertation erläutert. Abschnitt 2.1 gibt eine Einführung in die Kameraposeschätzung und Abschnitt 2.2 erläutert die Grundzüge der Graphentheorie als Modellierungswerkzeug. Einen kurzen Überblick über die Ausreißererkennung und über Relationen geben die Abschnitte 2.3 und 2.4. Klassifizierung mittels Random Forest wird im Abschnitt 2.5 und die Clusteranalyse im Abschnitt 2.6 behandelt.

2.1 Kameraposeschätzung

Kameraposeschätzung (auch: *Structure from Motion*) bezeichnet die Ermittlung von 3D-Punkten und Kameraposen (Abschnitt 2.1.1) anhand Merkmalskorrespondenzen zwischen Bildern in einer gegebenen (ungeordneten) Bildmenge. Dazu werden Merkmale in den Bildern detektiert und ihre Korrespondenzen zwischen den Bildern mittels Bildzuordnung (Abschnitt 2.1.2) bestimmt. Basierend auf den zugeordneten Merkmalen erfolgt die Bestimmung der Geometrie zwischen den Bildern (Abschnitte 2.1.3 bis 2.1.5) unter Beachtung von kritischen Kamerakonfigurationen (Abschnitt 2.1.6) und Bildschleifen (Abschnitt 2.1.7). Schließlich werden über die Vereinigung von Bildern (Abschnitt 2.1.8) die relativen Kameraposen für die gesamte Bildmenge ermittelt.

Dieser Abschnitt befasst sich mit der grundlegenden Theorie der Kameraposeschätzung und beschreibt relevante Verfahren. Eine ausführlichere Behandlung dieser Thematik erfolgt in den Büchern (Hartley und Zisserman 2004), (Faugeras et al. 2004), (Szeliski 2011), (McGlone 2013) sowie (Förstner und Wrobel 2016).

2.1.1 Kameramodell

Ein Kameramodell beschreibt, wie die dreidimensionale Szene durch die Kamera auf dem zweidimensionalen Bild dargestellt wird. Das einfachste und am häufigsten verwendete Kameramodell ist die Lochkamera (engl. *pinhole camera*). Die Abbildung der Szene mit den wichtigsten Elementen des Lochkameramodells sind in Abb. 2.1 dargestellt. Die Bildaufnahme entsteht durch die Zentralprojektion der dreidimensionalen Szene auf die zweidimensionale *Bildebene* (*fokale Ebene*). Das *Projektionszentrum* (*optisches Zentrum*, *Brennpunkt*) korrespondiert mit dem Loch der Kamera. Die Lichtstrahlen verlaufen vom 3D-Objekt aus auf das Projektionszentrum zu, wo sie sich kreuzen. Die *optische Achse* beschreibt eine Linie durch das Projektionszentrum, die orthogonal auf der Bildebene steht und diese im sogenannten *Bildhauptpunkt* schneidet. Die Distanz zwischen dem Projektionszentrum und der Bildebene wird als *Kamerakonstante* bezeichnet. In der Literatur wird diese auch häufig als *Brennweite* bezeichnet. Dies ist aber nicht

korrekt, da Brennweite ein Begriff aus der Optik ist, der streng genommen nur zutrifft, wenn die Kamera auf Unendlich fokussiert ist.

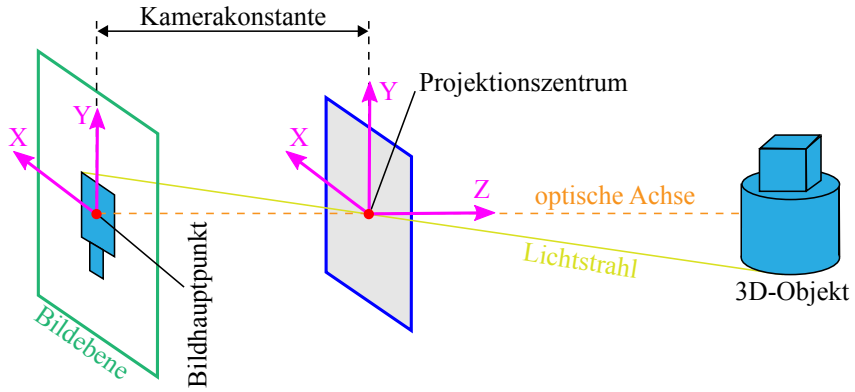


Abbildung 2.1: Lochkameramodell

Die Abbildung eines 3D-Punktes der Szene $\boldsymbol{X}$ auf einen 2D-Punkt in der Bildebene $\boldsymbol{x}$ erfolgt über die *Projektionsmatrix* $\boldsymbol{P}$ durch

$$\boldsymbol{x} = \boldsymbol{P} \cdot \boldsymbol{X} = \underbrace{\boldsymbol{K} \cdot \boldsymbol{K}_e}_{\boldsymbol{P}} \cdot \boldsymbol{X} \,, \tag{2.1}$$

wobei $\boldsymbol{X}$ und $\boldsymbol{x}$ in homogenen[1] Koordinaten gegeben sind. $\boldsymbol{P}$ ist eine homogene 3×4-Matrix, die sich aus einer extrinsischen Kameramatrix $\boldsymbol{K}_e$ und einer intrinsischen Kalibriermatrix $\boldsymbol{K}$ zusammensetzt. Die *extrinsische* 3×4-*Kameramatrix*[2]

$$\boldsymbol{K}_e = (\boldsymbol{R} \,|\, \boldsymbol{t}) \tag{2.2}$$

ist definiert durch die 3×3-Rotationsmatrix $\boldsymbol{R}$ und den Verschiebungsvektor $\boldsymbol{t}$ und beschreibt die *Kamerapose* in einem übergeordneten Koordinatensystem (*extrinsische Kameraparameter*). Die *intrinsische* 3×3-*Kalibriermatrix*

$$\boldsymbol{K} = \begin{pmatrix} f & s & h_x \\ 0 & f \cdot (1+m) & h_y \\ 0 & 0 & 1 \end{pmatrix} \tag{2.3}$$

beschreibt hingegen die inneren Eigenschaften der Kamera. Diese werden durch die Kamerakonstante f zur Beschreibung der Zentralprojektion, die Lage des Bildhauptpunktes (h_x, h_y), die Scherung des Kamerasensors s und den Faktor m zur Beschreibung von unterschiedlicher Skalierung in x- und y-Richtung charakterisiert (*intrinsische Kameraparameter*).

Die Lochkamera als ein ideales Modell kann die optischen Eigenschaften einer Kamera nur näherungsweise beschreiben. So können geometrische Abbildungsfehler der optischen Kamera

[1] *Homogene Koordinaten* $(x/w, y/w, z/w, w)$ werden in der projektiven Geometrie verwendet und erweitern kartesische Koordinaten (x, y, z) um eine Dimension in Form des *inversen Skalierungsfaktors* w. Zwei Punkte in homogenen Koordinaten repräsentieren denselben Punkt im euklidischen Raum genau dann, wenn einer ein Vielfaches des anderen ist und $w \neq 0$. Ein Punkt $(x, y, z, 0)$ repräsentiert einen Punkt im Unendlichen.

[2] $(\boldsymbol{M}|\mathbf{v})$ stellt eine 3×4-Matrix dar, die sich aus einer 3x3-Matrix $\boldsymbol{M}$ und einem dreidimensionalen Vektor $\mathbf{v}$ zusammensetzt.

zu einer lokalen Vergrößerung des Abbildungsmaßstabes zu den Rändern des Bildes hin führen. Dieser Effekt wird als *Verzeichnung* bezeichnet. Um diese auszugleichen, erfolgt die sogenannte *Verzeichnungskorrektur* eines Bildpunktes (u,v) auf einen nicht verzerrten Bildpunkt (x,y) durch

$$\begin{aligned} x &= u(1+k_1r^2+k_2r^4+...)+\Big(p_2(r^2+2u^2)+2p_1uv)\Big)\Big(1+p_3r^2+p_4r^4+...\Big) \\ y &= v(1+k_1r^2+k_2r^4+...)+\Big(p_1(r^2+2v^2)+2p_2uv)\Big)\Big(1+p_3r^2+p_4r^4+...\Big) \end{aligned} \tag{2.4}$$

mit k_i als dem i-ten *Radialverzeichnungsparameter*, p_i als dem i-ten *Tangentialverzeichnungsparameter* und $r=\sqrt{(u-h_x)^2+(v-h_y)^2}$, wobei (h_x,h_y) der Bildhauptpunkt ist.

Die Bestimmung der Kalibriermatrix sowie der Verzeichnungsparameter wird als *Kalibrierung* bezeichnet. Ist diese bekannt, spricht man von einer *kalibrierten* Kameraposeschätzung, ansonsten von einer *unkalibrierten*. Bei letzterer wird im Rahmen der sogenannten *Autokalibrierung* (Hartley und Zisserman 2004) die Kalibrierung automatisch während der Kameraposeschätzung bestimmt. Für eine zuverlässige Bestimmung der Kamerakonstante durch Autokalibrierung muss die aufgenommene Szene jedoch eine ausgeprägte 3D-Struktur besitzen. Bei unbekannten Bildmengen kann das jedoch nicht garantiert werden, weswegen oft Näherungswerte für die Kamerakonstante über die Metadaten der Bilddateien (z. B. Exchangeable Image File Format – Exif) ermittelt werden.

2.1.2 Bildzuordnung

Unter *Punkt-* oder *Bildzuordnung* versteht man die Bestimmung korrespondierender Merkmale zwischen Bildern. Zur Detektion der Merkmale wurde eine Vielzahl von Verfahren entwickelt, die sich grundlegend in Ecken- und Blobdetektoren unterteilen lassen (Krig 2014; Tuytelaars und Mikolajczyk 2008). Während Erstere Ecken detektieren, finden Letztere sogenannte Blobstrukturen, d. h. helle kreisförmige Regionen auf dunklem Hintergrund oder umgekehrt. Eckendetektoren wie der Förstner/Harris-Operator (Förstner und Gülch 1987; Harris und Stephens 1988) sind invariant gegenüber Verschiebungen und Rotationen, bestimmen jedoch lediglich Punkte auf einer Auflösung (Maßstab). Damit ist es nicht möglich, Merkmale in Bildern, die eine Szene in verschiedenen Maßstäben zeigen, zuverlässig zuzuordnen. Blobdetektoren wie Scale-Invariant Feature Transform – SIFT (Lowe 2004) oder Speeded Up Robust Features – SURF (Bay et al. 2006) sind im Gegensatz dazu in der Lage, auch die Größe eines Merkmals zu bestimmen, wodurch sie zusätzlich über eine Maßstabsinvarianz verfügen. Dies wird durch die Normierung des Maßstabs basierend auf der Laplace-Gaußfunktion erzielt (Lindeberg 1994).

Untersuchungen in (Mikolajczyk und Schmid 2005a) haben ergeben, dass sich SIFT-Merkmale am zuverlässigsten zuordnen lassen. Die Zuordnung basiert auf den sogenannten SIFT-Deskriptoren, die als 128-dimensionale Vektoren repräsentiert werden und mittels Euklidischer- oder Kosinusdistanz miteinander vergleichbar sind. Die Bestimmung der Deskriptoren erfolgt aus einem 16×16 großen Ausschnitt um das Merkmal basierend auf den Richtungen der darin enthaltenen Gradienten. Der Nachteil von SIFT im Vergleich zur Eckendetektoren ist der große Rechenaufwand zur Detektion von Merkmalen. Die notwendige Rechenzeit lässt sich jedoch mittels effizienter Implementierungen auf Grafikkarten (Wu 2012) deutlich reduzieren.

Größere Abstände der Kameras sowie Blickwinkelunterschiede führen zur stärkeren perspektivischen Verzerrungen zwischen den Aufnahmen. Robustere Operatoren wie Harris- und Hessian-Affine (Mikolajczyk und Schmid 2002; Mikolajczyk und Schmid 2005b) sowie Maximally Stable Extremal Regions – MSER (Matas et al. 2002) wurden für solche Fälle entwickelt. In (Morel und Yu 2009) wurde jedoch gezeigt, dass diese Operatoren zusammen mit SIFT-Deskriptoren unzureichend für starke perspektivische Bildverzerrungen sind. Es folgte die Entwicklung von robusteren Operatoren wie Affine-SIFT – ASIFT (Morel und Yu 2009) und Matching with On-Demand View Synthesis – MODS (Mishkin et al. 2015) sowie die Erweiterung von MODS (Roth et al. 2017), die alle auf der Simulation von Blickwinkelunterschieden basieren. Allerdings sind diese Ansätze deutlich aufwändiger und somit weniger geeignet für eine automatische Bildzuordnungen im Falle von größeren Bildmengen.

Einen Kompromiss zwischen Effizienz und Robustheit stellt die Bildzuordnung nach (Mayer et al. 2012) dar. Diese basiert auf SIFT-Merkmalen, verwendet jedoch anstatt der SIFT-Deskriptoren 13×13 Pixel große Ausschnitte um die Merkmale und (normierte) Kreuzkorrelation sowie kleinste Quadrate Zuordnung. Kreuzkorrelation hat dabei den Vorteil, dass gerade bei größeren Blickwinkeländerungen der Kreuzkorrelationskoeffizient nur graduell abnimmt, während der Vergleich der SIFT-Deskriptoren mehr oder weniger plötzlich zusammenbricht. Die Größe von 13×13 Pixel stellt einen Kompromiss dar zwischen erforderlicher Information und Reduktion der Störeinflüsse, die beispielsweise durch große Aufnahmeabstände entstehen können.

Um Zuordnung auch bei größeren Bildverzerrungen durchführen zu können, werden über eine Anpassung der SIFT-Parameter sehr viele Merkmale pro Bild extrahiert. Zur Reduktion des Suchraums für die möglichen Zuordnungen erfolgt eine Vorfilterung. Dazu werden die Ausschnitte mittels Kreuzkorrelation miteinander verglichen, wobei ein niedriger Schwellenwert von $0,5$ für den Kreuzkorrelationskoeffizienten verwendet wird.

Für die verbleibenden Zuordnungen wird eine kleinste Quadrate Bildzuordnung (Förstner 1984; Grün 1985) durchgeführt, bei der neben affinen geometrischen auch radiometrische Parameter bestimmt werden. Diesmal wird ein höherer Schwellenwert von $0,8$ für den Kreuzkorrelationskoeffizienten verwendet und zudem gefordert, dass die Varianz der Verschiebung einer gültigen Zuordnung unterhalb von $0,1$ Pixel liegt. Das Ergebnis ist neben den verbesserten Merkmalskoordinaten auch die zugehörige Kovarianzinformation. Diese kann als Fehlerellipse aufgefasst werden die angibt, wie genau die Zuordnung in unterschiedlichen Richtungen erfolgte.

2.1.3 Bündelausgleichung

Bündelausgleichung (Triggs et al. 1999; McGlone 2013; Förstner und Wrobel 2016) ist eine nichtlineare Methode zur gleichzeitigen Optimierung der Positionen der rekonstruierten 3D-Punkte $\boldsymbol{X}_j$ und der Kameraposen sowie deren internen Kalibrierparametern $\boldsymbol{P}_i$ in der Art, dass die Distanz zwischen den zurück projizierten $\boldsymbol{P}_i\boldsymbol{X}_j$ und den detektierten Merkmalspunkten $\boldsymbol{x}_j^i$ in den Bildern minimiert wird:

$$\min_{\boldsymbol{P}_i,\boldsymbol{X}_j} \sum_{ij} \mathrm{d}\left(\boldsymbol{P}_i\boldsymbol{X}_j,\boldsymbol{x}_j^i\right)^2 \tag{2.5}$$

Das Ziel besteht darin, die Fehler optimal auf alle Punkte zu verteilen. Die geometrische Distanz d entspricht hierbei einer nichtlinearen Modellfunktion deren Minimum mittels der Methode der kleinsten Quadrate über Normalgleichungen

$$\boldsymbol{N}\alpha = \left(\boldsymbol{A}^T\boldsymbol{A}\right)\alpha = \boldsymbol{A}^T\boldsymbol{y} \tag{2.6}$$

als $\alpha = \boldsymbol{N}^{-1}\boldsymbol{A}^T\mathbf{y}$ ermittelt wird. Die sogenannte *Designmatrix* $\boldsymbol{A}$ enthält partielle Ableitungen der Messungen bezüglich Kameras und 3D-Punkten, α die Parameter der Kameras und der 3D-Punkte und $\boldsymbol{y}$ die geometrischen Distanzen. Die Lösung über Normalgleichungen hat den Vorteil, dass die geschätzte Genauigkeitsinformation der detektierten Merkmalspunkte (vgl. Abschnitt 2.1.2) in Form der Kovarianzmatrix $\boldsymbol{C}$ als Gewichtung miteinbezogen werden kann:

$$\boldsymbol{N}\alpha = \left(\boldsymbol{A}^T\boldsymbol{C}^{-1}\boldsymbol{A}\right)\alpha = \boldsymbol{A}^T\boldsymbol{C}^{-1}\boldsymbol{y} \tag{2.7}$$

Dies wird im Rahmen der sogenannten *robusten Bündelausgleichung* (Mayer et al. 2012; McGlone 2013) ausgenutzt, bei der die Methode der kleinsten Quadrate in Form von M-Schätzern (Huber und Ronchetti 2009) verallgemeinert wird. Dazu wird die Distanz durch den Median aller Distanzen geteilt und eine Funktion des sich ergebenen Wertes zur zusätzlichen Gewichtung von $\boldsymbol{C}$ eingesetzt, was zur (iterativen) Regewichtung der Gleichung (2.7) führt. Hierbei werden 3D-Punkte, die stark abweichen, eliminiert, damit diese in der nächsten Iteration keinen negativen Einfluss mehr auf die restlichen Punkte haben. Die robuste Bündelausgleichung hat somit den Vorteil einer höheren Genauigkeit, allerdings mit etwas weniger 3D-Punkten.

Die mittlere Distanz d über alle 3D-Punkte wird als *mittlerer Rückprojektionsfehler* bezeichnet. Die Bündelausgleichung ist optimal bezüglich Minimierung dieses Rückprojektionsfehlers mittels Anpassung der 3D-Punkte und Kameraposen. Sie ist jedoch ein nicht lineares Optimierungsverfahren, das eine gute Startlösung für die 3D-Struktur und Kameraposen erfordert, um in die Nähe des globalen Optimums konvergieren zu können.

Die Bündelausgleichung ist ein relativ aufwändiges Optimierungsverfahren. Die Laufzeitkomplexität beträgt $\mathcal{O}((m+n)^3)$ pro Iteration mit einer Speicherkomplexität von $\mathcal{O}(mn(m+n))$, wobei m die Anzahl der Kameras und n die Anzahl der 3D-Punkte ist. Durch Ausnutzung der Eigenschaft, dass die Matrizen meist dünnbesetzt sind, lässt sich jedoch die Laufzeitkomplexität bei größeren realen Datensätzen auf $\mathcal{O}(m^3+mn)$ und die Speicherkomplexität auf $\mathcal{O}(mn)$ pro Iteration reduzieren (Mitra und Chellappa 2008).

2.1.4 Bildpaare

Ein *Bildpaar* (kurz: Paar) $P = (b_1, b_2)$ setzt sich aus zwei Bildern b_1 und b_2 zusammen, die den gleichen Teilbereich einer Szene abbilden. P wird hierbei als eine zweielementige Bildmenge betrachtet. Man spricht in diesem Fall auch davon, dass sich die Bilder *überlappen* bzw. einen *gemeinsamen Überlappungsbereich* aufweisen.

Zweibildgeometrie

Die relative Pose der Kameras der Bilder b_1 und b_2 wird durch die *Epipolargeometrie* beschrieben. In Abb. 2.2 sind die Elemente der Epipolargeometrie für eine konvergente Anordnung der Kameras dargestellt. Die Verbindungsgerade zwischen den beiden optischen Kamerazentren C_1 und C_2 wird als *Basis* und ihre Länge als *Basislänge* bezeichnet. Aufgrund der gedrehten Bildebenen schneidet die Basislinie die beiden Bildebenen. Diese Schnittpunkte werden *Epipole* $\boldsymbol{e}_1$ und $\boldsymbol{e}_2$ genannt und ihre Lage in den Bildebenen ist nur durch die Anordnung der Kameras zueinander bestimmt. Die Epipole können auch als Projektion der optischen Zentren C_1 und C_2 in die jeweils andere Bildebene aufgefasst werden. Ein 3D-Punkt $\boldsymbol{X}$ und die beiden Kamerazentren C_1 und C_2 spannen eine sogenannte *Epipolarebene* auf. Da die Abbildungen $\boldsymbol{x}_1$ und $\boldsymbol{x}_2$ von $\boldsymbol{X}$ auf den aus $\boldsymbol{X}$ und C_1 bzw. C_2 gebildeten Geraden liegen, sind auch diese Teil der Epipolarebene. Die Epipolarebene schneidet die beiden Bildebenen in zwei Schnittgeraden $\boldsymbol{l}_1$ und $\boldsymbol{l}_2$, die als *Epipolarlinien* bezeichnet werden.

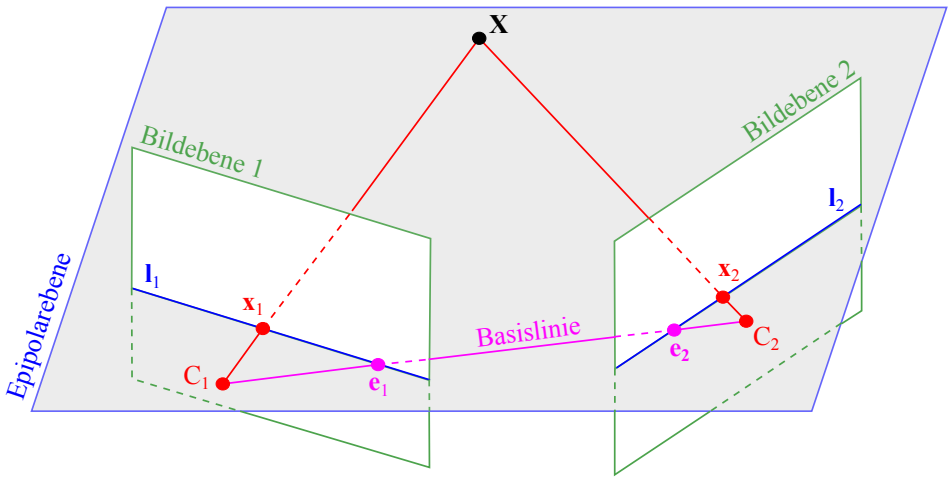


Abbildung 2.2: Elemente der Epipolargeometrie für konvergente Anordnung zweier Kameras mit den Zentren C_1 und C_2, bei der ein 3D-Punkt X auf die Bildpunkte x_1 und x_2 in den Bildebenen abgebildet wird.

Eine Bewegung des 3D-Punktes $\boldsymbol{X}$ innerhalb der Epipolarebene führt zu unterschiedlichen Abbildungen $\boldsymbol{x}_1$ und $\boldsymbol{x}_2$ in den beiden Bildebenen. Diese liegen jedoch immer auf den Epipolarlinien. Lässt man $\boldsymbol{X}$ entlang der Verbindungsgerade zwischen $\boldsymbol{X}$ und C_i in Richtung einer Kamera i laufen, so ergibt sich immer die gleiche Abbildung $\boldsymbol{x}_i$ in der Bildebene der Kamera i, während in der Bildebene der Kamera j die Abbildung $\boldsymbol{x}_j$ entlang der Epipolarlinie $\boldsymbol{l}_j$ in Richtung des Epipols $\boldsymbol{e}_j$ wandert. Diese Bewegung wird als *Disparität* bezeichnet. Die Verbindungsgerade zwischen $\boldsymbol{X}$ und C_i von jedem 3D-Punkt zu einer Kamera liefert als Projektion in der Bildebene der anderen Kamera die entsprechende Epipolarlinie. Folglich muss auch für jeden Bildpunkt in der Bildebene der einen Kamera in der Bildebene der anderen Kamera eine Epipolarlinie existieren, auf der alle möglichen Korrespondenzpunkte liegen.

Bei bekannten intrinsischen Kameraparametern erfolgt die Beschreibung der Epipolargeometrie über die sogenannte *essenzielle Matrix*. Ausgehend von einer konvergenten Kameraanordnung

mit Ursprung im Kamerakoordinatensystem der ersten Kamera ergeben sich die Projektionsmatrizen

$$\boldsymbol{P}_1 = \boldsymbol{K}_1 \cdot (\boldsymbol{I} \mid \boldsymbol{0}) \tag{2.8}$$

$$\boldsymbol{P}_2 = \boldsymbol{K}_2 \cdot \boldsymbol{R}(\boldsymbol{I} \mid -\boldsymbol{t}) \,, \tag{2.9}$$

wobei $\boldsymbol{K}_i$ die intrinsische Kameramatrix der i-ten Kamera und $\boldsymbol{R}(\boldsymbol{I}|-\mathbf{t})$ die extrinsische Matrix der zweiten Kamera bezüglich der ersten ist. Die essenzielle Matrix $\boldsymbol{E}$ kann durch

$$\boldsymbol{E} = (\boldsymbol{t})_\times \cdot \boldsymbol{R}^{-1} \tag{2.10}$$

beschrieben werden, wobei $()_\times$ für die schiefsymmetrische Matrix steht. Ohne Kenntnis der intrinsischen Kameraparameter verwendet man die sogenannte *Fundamentalmatrix*

$$\boldsymbol{F} = \boldsymbol{K}_2^{-T} \cdot \underbrace{(\boldsymbol{t})_\times \cdot \boldsymbol{R}^{-1}}_{\boldsymbol{E}} \cdot \boldsymbol{K}_1^{-1} \tag{2.11}$$

zur Beschreibung der Epipolargeometrie. Beide Matrizen befriedigen für jedes Paar von korrespondierenden Bildpunkten $\boldsymbol{x}_1$ und $\boldsymbol{x}_2$ bzw. kalibrierten Bildpunkten $\boldsymbol{x}_i^K$ in homogenen Bildkoordinaten die *Epipolargleichung*

$$\boldsymbol{x}_2^K \cdot \boldsymbol{E} \cdot \boldsymbol{x}_1^K = \boldsymbol{x}_2 \cdot \boldsymbol{F} \cdot \boldsymbol{x}_1 = 0 \,, \tag{2.12}$$

woraus sich ableiten lässt, dass korrespondierende Bildpunkte in einem Bild auf der entsprechenden Epipolarlinie im anderen Bild liegen müssen. Dadurch reduziert sich der zweidimensionale Suchraum auf einen eindimensionalen entlang der Epipolarlinie. Dieser Zusammenhang wird als *Epipolarbedingung* bezeichnet.

Die relative Pose (Rotation und Translation) zwischen Bildern eines Paares ist somit in der essenziellen Matrix enthalten. Im Gegensatz zur Rotation ist die Translation (Basislänge) mehrdeutig und lässt sich nur mit Zusatzwissen eindeutig definieren. Spezifischer enthält die essenzielle Matrix nur die Translationsrichtung, deren Länge üblicherweise auf 1 gesetzt wird.

Eine weitere Beziehung zwischen Punkten in zwei Bildebenen kann über die sogenannte *Homographie* hergestellt werden. Sie ist eine projektive Abbildung zwischen zwei Bildebenen. Das heißt, sie kann dazu verwendet werden, um Punkte einer Bildebene direkt auf die Punkte einer zweiten Bildebene abzubilden. Für ein Paar geschieht dies auf der Grundlage der sogenannten *Weltebene*. Man sagt, dass die Weltebene die Homographie zwischen den beiden Bildebenen *induziert*. Die Homographie $\boldsymbol{H}$ induziert durch die Ebene $\pi = (\boldsymbol{n}^T, d)^T$ ist gegeben durch

$$\boldsymbol{H} = \alpha \boldsymbol{K}_2 \left(\boldsymbol{R} - \frac{\boldsymbol{t}\boldsymbol{n}^T}{d} \right) \boldsymbol{K}_1^{-1} \,, \tag{2.13}$$

wobei $\boldsymbol{n}$ der Normalenvektor der Ebene, d der Abstand der Ebene vom Ursprung und α ein beliebiger Skalierungsfaktor ist. Die Beziehung zwischen den Bildpunkten ist dann über $\boldsymbol{x}_2 = \boldsymbol{H} \cdot \boldsymbol{x}_1$ gegeben und im Gegensatz zur Epipolargeometrie eindeutig.

Bestimmung der Zweibildgeometrie

Bei der Bestimmung der Epipolargeometrie wird zwischen direkten und robusten Verfahren unterschieden. *Direkte Verfahren* sind in der Lage, beim Vorliegen einer entsprechenden Anzahl an Merkmalskorrespondenzen, direkt (d. h. ohne Vorgabe einer Näherung) die Fundamentalmatrix bzw. die essenzielle Matrix zu bestimmen. Zu diesen Verfahren gehören neben dem 8-Punkt-Algorithmus (Longuet-Higgins 1981) und dem 7-Punkt-Algorithmus (Hartley und Zisserman 2004) auch der 5-Punkt-Algorithmus (Nistér 2004). Letzterer geht von bekannten intrinsischen Kameraparametern aus und ist somit zur direkten Bestimmung der essenziellen Matrix geeignet. Da häufig mehr als die Mindestanzahl an Korrespondenzen vorliegt, kann ein überbestimmtes lineares Gleichungssystem formuliert werden, das als Minimierungsproblem über die Methode der kleinsten Quadrate gelöst werden kann.

Direkte Verfahren gehen davon aus, dass korrekte sowie hinreichend genaue Merkmalskorrespondenzen vorliegen. Sie sind extrem empfindlich gegenüber Ausreißern in Form von falschen Merkmalskorrespondenzen, welche die Genauigkeit der geschätzten Epipolargeometrie hochgradig negativ beeinflussen können. Aus diesem Grund werden *robuste Verfahren* angewendet, die selbst beim Vorliegen einer großen Anzahl falscher Merkmalskorrespondenzen in der Lage sind, die Epipolargeometrie zu bestimmen. Diese basieren meistens auf dem nicht-deterministischen Schätzverfahren Random Sample Consensus (RANSAC) (Fischler und Bolles 1981) bzw. seinen Abwandlungen wie beispielsweise (Torr und Zisserman 2000) oder (Chum et al. 2003). Die grundlegende Vorgehensweise besteht dabei in der Kombination der Bestimmung der Epipolargeometrie mittels eines direkten Verfahrens und der zufälligen Auswahl von Merkmalskorrespondenzen. Basierend darauf werden korrekte Korrespondenzen (Konsens) ermittelt und ein bestimmtes Gütekriterium (beispielsweise die Zahl der korrekten Merkmalskorrespondenzen) bestimmt. Dies wird solange wiederholt, bis eine gewisse Anzahl von Iterationen erreicht ist. Die Anzahl der Iterationen kann fest vorgegeben werden oder aber aus dem momentan geschätzten Anteil an korrekten Punkten sowie einem vorgegebenen Signifikanzniveau (z. B. 99,9% für 1 Fehler bei 1000 Durchläufen) abgeleitet werden. Am Ende wählt man die Lösung aus, die zum besten Gütekriterium geführt hatte. Auf diese Weise wird erreicht, dass grob falsche Merkmalskorrespondenzen nicht in die Schätzung der Epipolargeometrie einfließen.

Eine höhere Robustheit und Genauigkeit lässt sich mit dem Verfahren von Mayer et al. (2012) erzielen. Es benutzt eine Strategie ähnlich zum Expectation-Maximization-Algorithmus (Dempster et al. 1977) zusammen mit RANSAC und dem Geometric Robust Information Criterion (GRIC) (Torr 1997) als Gütekriterium. Die Verwendung von GRIC ermöglicht, dass auch die Größe des Abstands zwischen Merkmalen und Epipolarlinien zur Bewertung der Lösung genutzt werden kann. Damit können Lösungen mit weniger Merkmalskorrespondenzen, die aber näher an der Epipolarlinie liegen, bevorzugt werden. Um lokale Lösungen, d. h. solche, die nur ein Teil eines Bildes beschreiben, zu vermeiden, werden die besten RANSAC-Lösungen mittels robuster Bündelausgleichung verbessert. Dies entfernt die Ausreißer basierend auf RANSAC sowie GRIC und optimiert zudem die Epipolargeometrie. Die optimierte Geometrie wird in der nächsten Iteration benutzt, um eine verbesserte Lösung zu generieren. Dies wird wiederholt, bis wiederum eine gewisse Anzahl an Iterationen erreicht ist oder sich der GRIC-Wert nicht mehr signifikant verbessert. Zuletzt wird die Lösung mit dem niedrigsten GRIC-Wert als Endergebnis ausgewählt.

2.1.5 Bildtriplets

Ein *Bildtriplet* (kurz: Triplet) $T = (b_1, b_2, b_3)$ setzt sich aus drei Bildern b_1, b_2 und b_3 zusammen, die den gleichen Teilbereich einer Szene abbilden. T wird hierbei als eine dreielementige Bildmenge betrachtet. In diesem Fall spricht man von einer *dreifachen Überlappung* zwischen Bildern bzw. einem *gemeinsamen Überlappungsbereich*.

Dreibildgeometrie

Die Kamerazentren der drei Bilder spannen die sogenannte *Trifokalebene* auf, die unabhängig von den 3D-Punkten ist (siehe Abb. 2.3). Die geometrischen Beziehungen zwischen den Bildern werden durch den *Trifokaltensor* (Hartley 1997; Torr und Zisserman 1997) beschrieben. Dieser setzt sich aus drei 3×3-Matrizen zusammen und hängt lediglich von der Lage der Bilder zueinander sowie den internen Kameraparametern ab. Somit ist er eindeutig durch die Projektionsmatrizen der Bilder definiert.

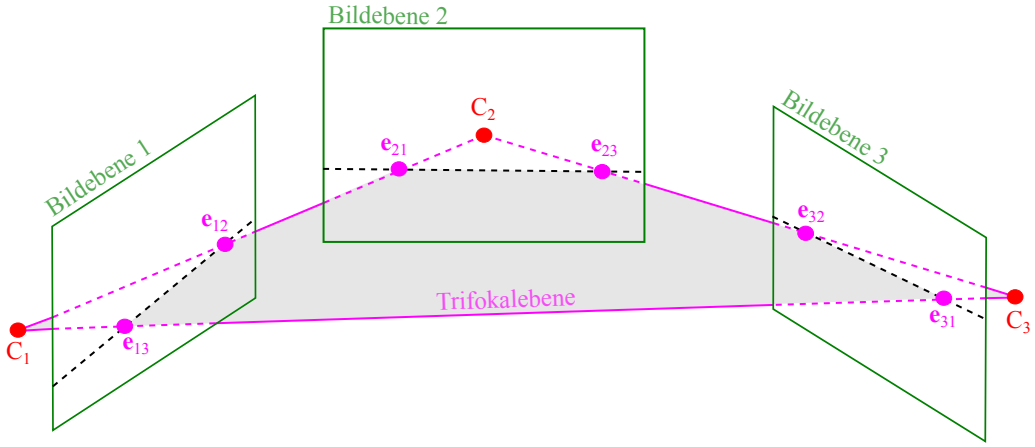


Abbildung 2.3: Dreibildgeometrie mit der Trifokalebene aufgespannt durch die Kamerazentren C_i, die die Epipole $\boldsymbol{e}_{ij}$ im Bild i bzgl. des Bildes j definieren.

Über den Trifokaltensor kann ein *Punkttransfer* erfolgen, bei dem ein Bildpunkt anhand der Korrespondenz in zwei Bildebenen direkt auf einen Bildpunkt in der dritten Bildebene transferiert wird. Dies vermeidet den aufwendigen Punkttransfer über die Projektionsmatrizen der Kameras.

Auf Grundlage des Punkttransfers ermöglichen Triplets eine eindeutige Überprüfung der Korrespondenzen, während diese bei Paaren nur auf die Epipolarlinie eingeschränkt werden können. Das führt letztendlich zu zuverlässigeren 3D-Punkten und Kameraposen (Fitzgibbon und Zisserman 1998).

Bestimmung der Dreibildgeometrie

Der Trifokaltensor wird meist nur im unkalibrierten Fall direkt geschätzt. Dazu können beispielsweise die in (Hartley und Zisserman 2004) beschriebenen Methoden eingesetzt werden. Bei

bekannter Kamerakalibrierung werden dagegen meist direkt die Projektionsmatrizen bestimmt, aus denen sich der Trifokaltensor ableiten lässt.

Für die Schätzung der Dreibildgeometrie werden mindestens vier Punkte benötigt, was allerdings komplexe und nicht symmetrische Verfahren wie beispielsweise (Nistér und Schaffalitzky 2006) erfordert. Stattdessen kann der in (Mayer et al. 2012) vorgestellte Ansatz verwendet werden, der auf zweimaliger Anwendung des 5-Punkt-Algorithmus beruht. Dazu wird eines der drei Bilder als Referenz (*Masterbild*) gewählt und für die anderen beiden Bilder (*Slavebilder*) in Bezug hierzu für die selben fünf Bildpunkte im Masterbild die relative Kamerapose bestimmt. Daraus ergeben sich relative Kameraposen, die bis auf die relative Skalierung eindeutig sind. Zur Bestimmung der Skalierung werden die verwendeten fünf Punkte trianguliert und die Verhältnisse der Abstände vom Masterbild berechnet. Die relative Skalierung ergibt sich robust als der Median der Verhältnisse. Eine höhere Robustheit für die Bestimmung der Dreibildgeometrie wird, wie in Abschnitt 2.1.4 für Paare beschrieben, durch eine Strategie ähnlich dem Expectation-Maximization-Algorithmus erzielt.

Die relativen Kameraposen (Rotationen und Translationen) der Bilder ergeben sich somit relativ zu einem Paar, in dem das Masterbild enthalten ist. Für dieses Paar kann die Basislänge auf 1 gesetzt werden (vgl. Abschnitt 2.1.4). Sei dies beispielsweise das Paar (b_1, b_2) mit dem Masterbild b_2. Die Kamerapose von Bild b_3 ergibt sich dann aus der essenziellen Matrix des Paares (b_2, b_3) und der bestimmten relativen Skalierung für die Basislänge.

2.1.6 Kritische Kamerakonfigurationen

Grundlegend wird bei der Bestimmung der Epipolargeometrie von einer allgemeinen Kamerabewegung und Szenenstruktur ausgegangen (Hartley und Zisserman 2004). Eine *allgemeine Kamerabewegung* bedeutet, dass die Kamera auf jeden Fall verschoben und eventuell rotiert wurde. Wenn die aufgenommene Szene in Relation zur Basis ein deutliches Relief besitzt, wird von einer *allgemeinen Szenenstruktur* gesprochen. Sind die genannten Bedingungen nicht erfüllt, so wird in Anlehnung an (Torr et al. 1999) zwischen folgenden *degenerierten Konfigurationen* unterschieden:

- *Strukturdegeneration*: Alle verfügbaren Merkmalskorrespondenzen bzw. die zugehörigen 3D-Punkte liegen auf einer Ebene.
- *Bewegungsdegeneration*: Die Kamera wurde nicht verschoben, sondern nur um ihr Projektionszentrum rotiert oder es wurden ihre internen Kameraparameter geändert.

Degeneration bedeutet hierbei, dass die Daten nicht genügend Bedingungen enthalten, um die Relation eindeutig zu bestimmen, sondern nur eine Klasse von Relationen. Im Falle von Strukturdegeneration sind die koplanaren Korrespondenzen unzureichend, um die Projektionsmatrix eindeutig zu bestimmen. Dies stellt jedoch nur ein Problem bei unkalibrierten Kameras und somit der Bestimmung der Fundamentalmatrix dar. Bei bekannter Kalibrierung kann die essenzielle Matrix mittels 5-Punkt-Algorithmus (Nistér 2004) selbst bei koplanaren Korrespondenzen bestimmt werden.

Die Bewegungsdegeneration ist hingegen weitaus problematischer und tritt sowohl im kalibrierten als auch im unkalibrierten Fall auf. Wurde die Kamera zwischen den Aufnahmen nicht bewegt, so ist die Epipolargeometrie undefiniert und die Bilder stehen über eine Homographie in Beziehung. Diese wird durch die Ebene im Unendlichen $\pi_\infty = (\boldsymbol{n}^T, \infty)^T$ induziert und als *unendliche Homographie*

$$\boldsymbol{H}_\infty = \lim_{d\to\infty} \boldsymbol{H} = \alpha \boldsymbol{K}_2 \boldsymbol{R} \boldsymbol{K}_1^{-1} \tag{2.14}$$

bezeichnet. Sie ergibt sich daraus, dass der Term $\boldsymbol{t}\boldsymbol{n}^T/d$ in Gleichung (2.13) 0 wird, wenn die Translation $\boldsymbol{t} = \boldsymbol{0}$ ist, was einer reinen Rotation der Kamera um ihr Zentrum entspricht. Das bedeutet, dass $\boldsymbol{H}_\infty$ unabhängig von der Translation $\boldsymbol{t}$ zwischen den Kameras ist und nur von der Rotation und den internen Kameraparametern abhängt. Dieser Zusammenhang wurde beispielsweise in (Agapito et al. 2001; Hartley et al. 1999) zur Bestimmung der intrinsischen Kameraparameter ausgenutzt.

Degenerierte Kamerakonfigurationen beziehen sich auf die notwendigen geometrischen Bedingungen zur Schätzung der Epipolargeometrie. Aufgrund von Ungenauigkeiten in der Lokalisierung der Merkmale und der Kalibrierung kann es aber auch bei allgemeiner Kamerabewegung und Szenenstruktur zu falsch geschätzter Epipolargeometrie kommen. Dies hängt vom Verhältnis der Basislänge zum Abstand der Szenenobjekte ab. Insbesondere die Bestimmung der Translationsrichtung ist demgegenüber empfindlich. Im Gegensatz dazu kann die Rotation selbst für kleine Basislängen zuverlässig bestimmt werden (Enqvist et al. 2011). Unter *kritischen Kamerakonfigurationen* werden hier somit neben reinen Rotationen auch Konfigurationen mit einem geringen Verhältnis der Basislänge zum Abstand der Szenenobjekte verstanden.

Während die 3D-Rekonstruktion im Falle der reinen Rotation nicht definiert ist, erfordert eine zuverlässige Schätzung von 3D-Geometrie, Kameraposen sowie intrinsischen Kameraparametern eine ausreichende Basis zwischen den Bildern. Ein Gütemaß hierfür wird im weiteren Verlauf als *Stabilität* bezeichnet, wobei eine höhere Stabilität eine bessere Schnittgeometrie impliziert. Mit steigender Basislänge erhöhen sich jedoch die perspektiven Verzerrungen zwischen den Bildern sowie die Wahrscheinlichkeit von Verdeckungen. Letztere können wiederum zum Verlust von Korrespondenzen und aufgrund der dadurch reduzierter Redundanz zu instabiler Schätzung führen (Triggs et al. 1999). Aus diesen Gründen muss ein Kompromiss zwischen ausreichender Basislänge (Stabilität) und der Korrespondenzanzahl gefunden werden. Die Charakterisierung hiervon erfolgt über die sogenannte *Qualität* als Gütemaß.

2.1.7 Schleifenschluss

Eine *Bildsequenz* bezeichnet sequenziell angeordnete Bilder mit paarweisen Verknüpfungen. Bilder die sich an den Enden der Bildsequenz befinden werden *Endbilder* genannt. Überlappen sich diese, so handelt es sich bei der Bildsequenz um eine sogenannte *Bildschleife*. Unter *Schleifenschluss* versteht man die Verknüpfung der Endbilder einer Bildsequenz, um eine Bildschleife zu erhalten.

Ungenauigkeiten in den 3D-Punkten können sich während der Vereinigung von Bildteilmengen (Abschnitt 2.1.8) aufsummieren und zu signifikanten Abweichungen der geschätzten von den tatsächlichen Kamerapositionen führen. Im unkalibrierten Fall spricht man hierbei von *projektiver Abweichung* (engl. *projective drift*). Die Ungenauigkeiten in den 3D-Punkten entstehen vor allem

durch ungenaue Lokalisierung von Merkmalspunkten sowie Näherungswerte für die intrinsischen Kameraparameter. Am deutlichsten ist dieser Effekt bei langen nicht geschlossenen Bildschleifen ausgeprägt. Hierbei stimmen die Kamerapositionen der Endbilder überein, die geschätzten weichen jedoch, aufgrund der Aufsummierung von Fehlern, unter Umständen deutlich voneinander ab.

In (Steedly et al. 2003) wurde gezeigt, dass bei langen Bildsequenzen signifikante Abweichungen entlang des Pfades lediglich zu kleinen Fehlern in den Bildern führen. Daraus ergibt sich, dass in solchen Fällen die Kameraposen nicht ohne einen expliziten Schleifenschluss genau bestimmt werden können. Bei kurzen Bildsequenzen hingegen führt die Aufsummierung von Ungenauigkeiten zu vernachlässigbaren Abweichungen in der 3D-Geometrie (Repko und Pollefeys 2005).

2.1.8 Vereinigung von Bildteilmengen

Paare bzw. Triplets stellen *Bildteilmengen* der zugrunde liegenden Bildmenge dar und werden hier als *Basisblöcke* einer Kameraposeschätzung bezeichnet. Eine *Vereinigung* von zwei Bildteilmengen mit n und m sowie k gemeinsamen Bildern führt zu einer Bildteilmenge mit $n+m-k$ Bildern, sowie zu relativen Kameraposen für alle Bilder der gebildeten Bildteilmenge. In diesem Zusammenhang versteht man unter *relativen Kameraposen* die Kameraposen der Bilder relativ zu einem Bild aus der gleichen Bildteilmenge, das sich im Ursprung befindet.

Über eine wiederholte Vereinigung kann die Bestimmung von Kameraposen innerhalb einer Bildmenge erfolgen. Dabei wird zwischen inkrementeller und hierarchischer Vereinigung unterschieden. Die *inkrementelle* oder *sequenzielle Vereinigung* (Snavely et al. 2006; Agarwal et al. 2009; Frahm et al. 2010; Mayer et al. 2012; Heinly et al. 2015; Schönberger und Frahm 2016) startet mit einem initialen Basisblock. Anschließend erfolgt eine iterative Vereinigung der restlichen Basisblöcke, wodurch aus dem initialen Basisblock eine immer größere Bildteilmenge wächst. Praktische Erfahrung zeigt, dass für eine robuste Lösung das Ergebnis jeder Vereinigung mittels Bündelausgleichung optimiert werden muss, was zu einer exponentiell steigenden Komplexität führt.

Bei der *hierarchischen Vereinigung* (Fitzgibbon und Zisserman 1998; Farenzena et al. 2009; Gherardi et al. 2010; Mayer 2014; Toldo et al. 2015) erfolgt hingegen zunächst eine Unterteilung der Menge der Basisblöcke in unabhängige Teilmengen, deren Elemente anschließend inkrementell vereinigt werden. Die resultierenden Bildteilmengen werden als neue Basisblöcke betrachtet und erneut hierarchisch vereinigt. Diese Vorgehensweise wiederholt sich bis eine Bildteilmenge, die der Bildmenge entspricht, erreicht ist.

Die inkrementelle Vereinigung weist im Vergleich zur hierarchischen einige Nachteile auf:

- Die Wahl des initialen Basisblocks beeinflusst die Genauigkeit der Kameraposeschätzung.
- Aufsummierung von Fehlern führt zu einer stärkeren geometrischen Deformation (vgl. Abschnitt 2.1.7).
- Die einzelnen Vereinigungsschritte sind nicht unabhängig voneinander.

Die Bündelausgleichung als nicht lineares Optimierungsverfahren erfordert eine Startlösung, die sich nahe genug am globalen Optimum befinden muss. Ansonsten konvergiert sie zu einem lokalen Optimum, was zu ungenauen Kameraposen oder gescheiterter Vereinigung führt. Der initiale Basisblock mit seiner rekonstruierten 3D-Struktur sowie seinen geschätzten Kameraposen stellt die Startlösung für die Bündelausgleichung dar und beeinflusst die Endlösung.

In der Regel ist die Bündelausgleichung in der Lage, die Fehler gleichmäßig zu verteilen. Wenn jedoch der initiale Basisblock eine Startlösung darstellt, die zu weit weg vom globalen Optimum ist, konvergiert die Bündelausgleichung unter Umständen zu einem schlechten, lokalen Optimum. Dies lässt sich mittels hierarchischer Vereinigung weitgehend reduzieren, da hierbei die Fehler besser verteilt werden (Hartley und Zisserman 2004).

Der größte Nachteil der inkrementellen Vereinigung ist aber ihre sequenzielle Ausführung aufgrund der Abhängigkeit zwischen den einzelnen Vereinigungsschritten. Zusammen mit der Tatsache, dass nach jeder Vereinigung das Ergebnis mittels Bündelausgleichung aufwändig optimiert wird, macht das die inkrementelle Vereinigung außerordentlich langsam. Im Gegensatz dazu sind bei der hierarchischen Vereinigung die einzelnen Vereinigungsschritte unabhängig voneinander. Auf parallelen Rechnerarchitekturen kann dies ausgenutzt werden, um die Effizienz der Kameraposeschätzung signifikant zu erhöhen.

Eine Alternative stellen *globale Verfahren* dar, die erst relative Rotationen zwischen Paaren bzw. Triplets bestimmen, dann die Translationen schätzen und dies alles anschließend mittels Bündelausgleichung optimieren (Martinec und Pajdla 2007; Crandall et al. 2011; Moulon et al. 2013; Wilson und Snavely 2014; Sweeney et al. 2015; Reich und Heipke 2016). Auf diese Weise werden mehrere aufwändige Bündelausgleichungen sowie Probleme mit dem initialen Basisblock vermieden. Allerdings sind diese Verfahren empfindlich gegenüber Ausreißern in Form von inkonsistenten Geometrien, da diese nicht wie bei den inkrementellen oder hierarchischen Verfahren herausgefiltert werden können.

2.2 Graphentheorie

Ein *ungerichteter Graph* (kurz: Graph) ist ein Tupel $G = (V, E)$, wobei V eine Menge von *Knoten* und E eine Menge von *Kanten* bezeichnet. Jede (ungerichtete) Kante $e = (u, v) \in E$ verbindet zwei Knoten $u \in V$ und $v \in V$, die als *adjazent* (*benachbart*) bezeichnet werden. Ein Knoten u und eine Kante e heißen *inzident*, wenn e den Knoten u mit einem anderen Knoten aus V verbindet. Zwei Kanten heißen *benachbart*, wenn sie mit dem gleichen Knoten inzident sind. Ansonsten bezeichnet man sie als *unabhängig*. Die Anzahl der Kanten, die mit einem Knoten inzident sind, entspricht dem *Grad* des Knotens.

Ein *gewichteter Graph* ist ein Tupel $G = (V, E, w)$ mit der *Kantengewichtsfunktion* $\mathrm{w} : E \to \mathbb{R}$, die jeder Kante eine reelle Zahl als *Kantengewicht* zuordnet. Ein Graph wird als *metrisch* bezeichnet, wenn w eine Metrik ist und somit die Dreiecksungleichung

$$\mathrm{w}(a, c) \leq \mathrm{w}(a, b) + \mathrm{w}(b, c) \qquad \forall a, b, c \in E \tag{2.15}$$

erfüllt. Das bedeutet, dass der Weg von a über b nach c keine geringere Summe der Kantengewichte haben darf als der direkte Weg von a nach c.

Das Verhältnis von tatsächlich vorhandenen zu potentiell möglichen Kanten

$$\frac{2|E|}{|V|(|V|-1)} \tag{2.16}$$

bezeichnet man als die *Dichte* von G. Ist jeder Knoten des Graphen mit jedem anderen verbunden, heißt ein Graph *vollständig*, was einer Dichte von 1 entspricht. Ein Graph mit einer Dichte nahe 1 wird *dichter Graph* genannt.

2.2.1 Teilgraphen

Ein *Teilgraph* $G_T = (V_T, E_T)$ eines Graphen $G = (V, E)$ ist ein Graph mit $V_T \subseteq V$ und $E_T \subseteq E$. Weiter wird zwischen einem *induzierten* Teilgraphen, der alle Kanten aus E enthält, die mit Knoten aus V_T inzident sind, und einem *aufspannendem* Teilgraphen oder *Faktor* mit $V_T = V$ unterschieden. Bei gewichteten Graphen wird zudem gefordert, dass die Kantengewichtsfunktion übereinstimmt.

Ein vollständiger Teilgraph G_T wird als $|V_T|$-*Clique* bezeichnet. Die Entscheidung, ob ein Graph eine Clique einer bestimmten Mindestgröße enthält, wird *Cliquenproblem* genannt und gilt als NP-vollständiges Problem.

2.2.2 Zusammenhang

Ein Graph heißt *zusammenhängend*, falls von jedem seiner Knoten jeder andere erreichbar ist. Die Teilgraphen eines nicht zusammenhängenden Graphen werden *Zusammenhangskomponenten* genannt.

Eine *Artikulation* in einem zusammenhängenden Graphen ist ein Knoten, dessen Elimination die Anzahl der Zusammenhangskomponenten erhöht. Übertragen auf Kanten spricht man in diesem Fall von einer *Brücke*.

2.2.3 Wege und Pfade

Ein *Pfad* (engl. *path*, *simple path*) in einem Graphen ist eine Folge von Knoten oder Kanten ohne Wiederholung. Ein *Weg* (engl. *walk*) ist hingegen eine Folge von Knoten oder Kanten, bei der Wiederholungen von Knoten oder Kanten möglich sind. Somit ist jeder Pfad in einem Graphen auch ein Weg.

Ein Pfad, bei dem Start- und Endknoten identisch sind, heißt *Kreis* und ein Weg für den das Gleiche gilt *Zyklus*. Ein Graph mit mindestens einem Zyklus heißt *zyklisch*, ansonsten *azyklisch*. Ein Zyklus oder Kreis heißt *trivial*, wenn er weniger als drei Knoten enthält. Triviale Kreise oder Zyklen werden bei der Analyse von Graphen meist nicht betrachtet. Einen Kreis, der genau drei Knoten enthält, bezeichnet man als *Dreieck* (3er-Clique). Einen Graphen ohne Dreiecke nennt man *dreiecksfrei*.

Die Summe der Kantengewichte entlang eines Pfads nennt man *Pfadlänge*. Der kürzeste Pfad zwischen zwei Knoten wird als ihr *Abstand* bezeichnet. Die *Exzentrizität* eines Knotens ist der maximale Abstand dieses Knotens von jedem anderen Knoten im Graphen.

2.2.4 Bäume

Ein *Baum* ist ein zusammenhängender, azyklischer Graph $B = (V, E)$ mit $|E| = |V| - 1$ Kanten. Knoten, die den Grad 1 besitzen, werden als *Blätter* oder *äußere Knoten* des Baumes bezeichnet und alle anderen als *innere Knoten*. Die Kanten eines Baumes nennt man auch *Verzweigungen*.

Ein Baum ist minimal zusammenhängend, d. h. er zerfällt in Zusammenhangskomponenten, wenn man eine beliebige Kante entfernt. Somit entspricht jede Kante einer Brücke. Das Einfügen einer Kante zwischen beliebigen Knoten führt hingegen zu genau einem Kreis im Baum.

Ein besonderer Knoten in einem Baum wird als *Wurzel* und der zugehörige Baum als *gewurzelt* bezeichnet. Hierbei entspricht die *Tiefe* (*Ebene*) eines Knotens seinem Abstand von der Wurzel. Die *Höhe* eines gewurzelten Baumes ist die Länge des längsten Pfades von der Wurzel und entspricht somit der größten Tiefe.

Ein *Binärbaum* ist ein gewurzelter Baum, bei dem der Wurzelknoten den Grad 2 besitzt und alle anderen Knoten entweder den Grad 1 oder 3.

Spannbaum

Ein *Spannbaum* eines Graphen $G = (V, E)$ ist ein baumförmiger aufspannender Teilgraph $S(G) = (V, E_S)$ mit $E_S \subseteq E$. Ein Spannbaum existiert nur in zusammenhängenden Graphen. In nicht zusammenhängenden Graphen existiert für jede Zusammenhangskomponente ein Spannbaum, die zusammen einen *Spannwald* definieren.

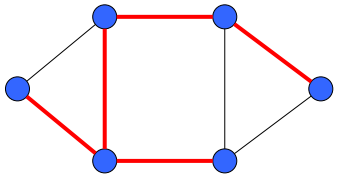


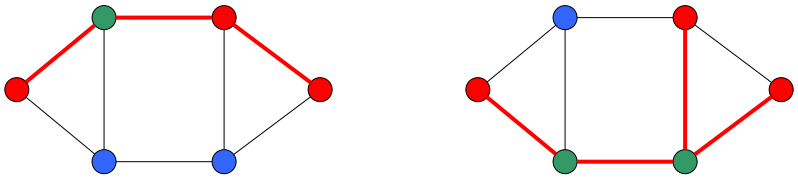
Abbildung 2.4: Spannbaum mit zugehörigen Kanten dargestellt in rot.

In gewichteten Graphen ist beim *minimalen Spannbaum* die Summe der Kantengewichte minimal. Das heißt, es existiert kein anderer Spannbaum im Graphen mit einer niedrigeren Summe der Kantengewichte. Ein minimaler Spannbaum ist eindeutig, wenn alle Kanten im Graphen unterschiedliche Gewichte aufweisen. Ansonsten können mehrere minimale Spannbäume existieren. Im Fall von nicht zusammenhängenden Graphen spricht man vom *minimalen Spannwald*.

Ein Spannbaum lässt sich mittels Breiten- oder Tiefensuche ermitteln (Cormen et al. 2013). Zur Bestimmung eines minimalen Spannbaums werden am häufigsten die Algorithmen von Kruskal (Kruskal 1956) oder Prim (Prim 1957) eingesetzt.

Steinerbaum

Ein *Steinerbaum* eines Graphen $G = (V, E)$ ist ein baumförmiger aufspannender Teilgraph $St(G) = (V_{St}, E_{St})$ mit $V_{St} \subseteq V$ und $E_{St} \subseteq E$. Die Knotenmenge $V_{St} = T \cup N$ setzt sich aus der Menge der *Terminale* $T \subseteq V$ und der Menge der *Steinerknoten* $N = V \backslash T$ zusammen. In einem Steinerbaum müssen alle Terminale und es können beliebig viele Steinerknoten enthalten sein. Letztere werden hinzugefügt, um Verbindungen zwischen Terminalen herzustellen. Wird zusätzlich gefordert, dass Terminale ausschließlich als Blätter des Baumes vorkommen dürfen, spricht man von einem *terminalen Steinerbaum* (Lin und Xue 2002). In Abb. 2.5 sind ein Steinerbaum sowie der terminale Steinerbaum für die gleiche Terminalmenge dargestellt.



(a) Steinerbaum (b) Terminaler Steinerbaum

Abbildung 2.5: Steinerbäume mit zugehörigen Kanten und Terminalen dargestellt in rot sowie Steinerknoten in grün.

Der Steinerbaum wurde erstmals in (Hakimi 1971) formuliert und stellt eine Verallgemeinerung des Spannbaums dar. Für $V_{St} = V$ stimmt ein Steinerbaum mit einem Spannbaum überein. Äquivalent zu den Spannbäumen sucht man in gewichteten ungerichteten Graphen nach *minimalen Steinerbäumen*, die ebenfalls mehrdeutig sein können.

Die Suche nach einem Steinerbaum wird als *Steinerbaumproblem* bezeichnet und gehört zur Klasse der NP-vollständigen Probleme (Lin und Xue 2002). Sie bleibt NP-vollständig, selbst wenn die Kantengewichte gleich sind, G planar[3] oder bipartit[4] ist. Somit existiert kein Algorithmus, der eine polynomiale Laufzeit in Abhängigkeit von der Knotenanzahl besitzt. Zwei Spezialfälle können jedoch in polynomialer Zeit gelöst werden:

- Für $T = V$ handelt es sich um die Suche nach einem minimalen Spannbaum.
- Für $|T| = 2$ lässt sich das Problem auf die Suche nach einem kürzesten Pfad zwischen den beiden Terminalen reduzieren.

Für das Steinerbaumproblem existiert daher neben exakten Verfahren (Fuchs et al. 2007; Dreyfus und Wagner 1971; Winter 1987) auch eine Vielzahl von Näherungsverfahren (Robins und A. Zelikovsky 2005; Hougardy und Prömel 1999; A. Z. Zelikovsky 1993; Kou et al. 1981; Mehlhorn 1988; Winter 1987), die für praktische Anwendungen ausreichende Näherungen liefern.

[3] Ein *planarer Graph* ist ein Graph, dessen Knoten und Kanten sich als Punkte und Linien in einer Ebene darstellen lassen ohne dass sich die Kanten schneiden.

[4] Ein *bipartiter Graph* ist ein Graph, dessen Knotenmenge in zwei disjunkte Teilmengen unterteilbar ist, sodass zwischen den Knoten der beiden Teilmengen keine Kanten verlaufen.

Die Suche nach einem terminalen Steinerbaum stellt ein noch schwierigeres Problem dar. Es ist ebenfalls NP-vollständig und die Approximation auf eine konstante Approximationsgüte kann nur für metrische Graphen gegeben werden (Lin und Xue 2002; Drake und Hougardy 2004). Auch hierfür wurden verschiedene Näherungsverfahren (Lin und Xue 2002; Fuchs 2003; Drake und Hougardy 2004; Martinez et al. 2007; Chen 2011) entwickelt, um das Problem effizienter zu lösen.

2.2.5 Kantengraph

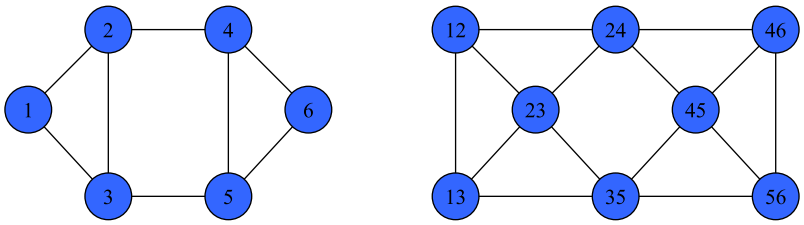
Der *Kantengraph* $L(G) = (V_L, E_L)$ eines ungerichteten Graphen $G = (V, E)$ besitzt als Knotenmenge die Kanten von G. Zwei Knoten in $L(G)$ heißen *benachbart*, wenn sie genau einen Knoten von G gemeinsam haben. Die Adjazenzmatrix $\boldsymbol{A}_{L(G)}$ (Cormen et al. 2013) des Kantengraphen ist definiert als

$$\boldsymbol{A}_{L(G)} = \boldsymbol{R}_G^T \boldsymbol{R}_G - 2\boldsymbol{I}, \tag{2.17}$$

wobei $\boldsymbol{I}$ die Einheitsmatrix und $\boldsymbol{R}_G$ die $|V| \times |E|$-Inzidenzmatrix (Cormen et al. 2013) von G ist. Ein Element in $\boldsymbol{R}_G$ hat den Wert 1, wenn die zugehörige Kante und Knoten inzident sind, ansonsten hat das Element den Wert 0. Die Anzahl der Knoten in $L(G)$ entspricht der Anzahl der Kanten in G, d. h. $|V_L| = |E|$. Die Anzahl der Kanten im Kantengraphen ergibt sich durch

$$|E_L| = \frac{1}{2} \sum_{n \in V} \mathrm{d}_n^2 - |E|, \tag{2.18}$$

wobei d_n der Grad des Knotens n in G ist. Daraus folgt, dass jeder Knoten n in G genau d_n Knoten bzw. eine d_n-Clique in $L(G)$ generiert. Somit führt jeder Knoten n zu $\binom{d_n}{2}$ neuen Kanten in $L(G)$.



(a) Graph (b) Kantengraph

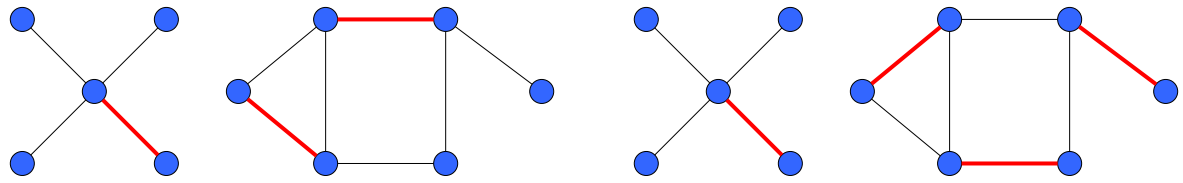
Abbildung 2.6: Kantengraph eines Graphen

2.2.6 Paarung

Eine *Paarung* (engl. *matching*) $M \subseteq E$ eines Graphen $G = (V, E)$ ist eine Teilmenge seiner paarweise nicht benachbarten Kanten. Ihre *Größe* entspricht der Anzahl der enthaltenen Kanten. Eine Paarung heißt

- *nicht erweiterbar* (engl. *maximal*), falls es keine Kante $e \in E$ gibt, sodass $M \cup \{e\}$ eine Paarung bleibt (siehe Abb. 2.7(a)). Das bedeutet, dass keine Kante aus E, die noch nicht in M enthalten ist, existiert und zu M hinzugefügt werden kann, ohne dass M keine Paarung mehr ist.
- *maximal* (engl. *maximum*), falls sie die größtmögliche Anzahl der Kanten enthält. Das heißt, es existiert keine Paarung mit mehr Kanten (siehe Abb. 2.7(b)). Es können viele maximale Paarungen existieren. Zu beachten ist, dass jede maximale Paarung nicht erweiterbar ist, aber die Umkehrung nicht notwendigerweise gilt.
- *perfekt* (engl. *complete*, *1-factor*), falls jeder Knoten aus V mit exakt einer Kante aus M inzident ist. Eine perfekte Paarung enthält somit $n/2$ Kanten, wobei n die Anzahl der Knoten ist. Zu beachten ist, dass die perfekte Paarung nur bei Graphen mit gerader Anzahl der Knoten möglich ist. Außerdem besitzen nicht alle Graphen eine perfekte Paarung, jedoch eine maximale. Ein Graph besitzt entweder die gleiche Anzahl an perfekten wie maximalen Paarungen oder keine perfekte Paarung. In Abb. 2.7(b) besitzt der rechte Graph eine perfekte Paarung, der linke aber nicht. Jede perfekte Paarung ist jedoch maximal und somit nicht erweiterbar.
- *beinahe perfekt*, falls genau ein Knoten nicht überdeckt werden kann. Dies ist genau dann der Fall, wenn der Graph eine ungerade Anzahl an Knoten besitzt.

Ein Knoten ist *überdeckt* (engl. *matched*, *covered*, *saturated*), wenn eine der inzidenten Kanten in der Paarung enthalten ist, andernfalls wird der Knoten als *nicht überdeckt* (engl. *unmatched*, *exposed*) bezeichnet.



(a) Nicht erweiterbare Paarungen (b) Maximale Paarungen

Abbildung 2.7: Paarungsarten mit überdeckten Kanten dargestellt in rot.

Jeder zusammenhängender Graph mit gerader Anzahl an Knoten und ohne sternförmige Verbindungen (engl. *claw-free*) besitzt eine perfekte Paarung (David 1974; Michel 1975). Eine sternförmige Verbindung (engl. *claw*) entspricht beispielsweise dem linken Graphen in Abb. 2.7(a). Ob ein Graph eine sternförmige Verbindung besitzt, kann in $O(|E|^4)$ getestet werden, indem jedes 4-Tupel von Knoten auf eine sternförmige Verbindung überprüft wird.

Im Falle eines gewichteten Graphen kann auch nach einer Paarung mit dem maximalen/minimalen Gesamtgewicht der Kanten gesucht werden. Dazu wird für allgemeine Graphen meistens der Algorithmus von Edmonds (Edmonds 1965) verwendet.

2.3 Ausreißererkennung

Als *Ausreißer* wird ein Element bezeichnet, das deutlich von den anderen Elementen aus der gleichen Elementmenge abweicht (Grubbs 1969). Andernfalls spricht man von einem *Inlier*. Um Ausreißer zu erkennen wurde eine Vielzahl von Verfahren entwickelt (Hodge und Austin 2004; Zhang 2013). Die Robustheit eines Schätzers gegenüber Ausreißern wird über den *Bruchpunkt* (Hampel 1971) charakterisiert. Dieser entspricht dem minimalen Anteil an kontaminierten Daten, der nötig ist, um das Ergebnis des Schätzers beliebig zu verfälschen. Im Allgemeinen gilt, dass je größer der Bruchpunkt ist, desto robuster ist der Schätzer.

Eine statistische Vorgehensweise, um Ausreißer zu erkennen, besteht darin anzunehmen, dass die Elemente näherungsweise normalverteilt sind. Die Wahrscheinlichkeit Elemente, die weiter als das Dreifache der Standardabweichung vom Mittelwert liegen, zu beobachten liegt dann bei etwa 0.3%. Allerdings ist die Schätzung des Mittelwerts und der Varianz aus kontaminierten Daten problematisch, weil der Schätzer den Bruchpunkt 0 besitzt. Wenn beispielsweise nur ein Element den Wert ∞ besitzt (*Hebelpunkt*), führt dies dazu, dass sowohl der Mittelwert als auch die Varianz ∞ sind.

Ein deutlich robusterer Schätzer ist hingegen der Median mit dem Bruchpunkt von $0,5$, insbesondere der Median der absoluten Abweichungen

$$\mathrm{MAD} = \underset{e \in E}{\mathrm{Median}} \{ |e - \mathrm{Median}\{E\}| \} \tag{2.19}$$

bezüglich der Elemente $e \in E$ einer Elementmenge E. Die Standardabweichung von normalverteilten Elementen entspricht in diesem Fall 1.4826 MAD. Damit ist eine robuste Ausreißererkennung mittels der sogenannten *X84-Regel* (Hampel et al. 2005) möglich. Bei dieser werden Daten, die mehr als $1,4826k$ MAD vom Median entfernt sind, als Ausreißer detektiert. Der Parameter k entspricht hierbei der k-fachen Standardabweichung. Diese Regel hat sich als sehr effizient herausgestellt (Pearson 2002) und wurde bereits in einigen praktischen Anwendungen, wie beispielsweise in (Fusiello et al. 2002) oder (Kangni und Laganiére 2007) erfolgreich eingesetzt.

2.4 Relationen

Eine *Relation* ist eine Beziehung zwischen zwei Elementen, die entweder bestehen kann oder nicht. Mengentheoretisch lässt sie sich als eine Menge von Tupeln definieren, wobei zwischen den Elementen eines Tupels eine Relation besteht. Eine zweistellige Relation R zwischen zwei Elementen x und y kann als xRy geschrieben werden. Sie wird als *Äquivalenzrelation* bezeichnet, wenn folgende Eigenschaften gegeben sind:

- *Reflexivität*: Jedes Element x steht in Relation zu sich selbst, d. h. xRx.
- *Symmetrie*: Aus der Relation xRy folgt immer yRx, d. h. $xRy \Rightarrow yRx$.
- *Transitivität*: Aus der Relation xRy und xRz folgt immer yRz, d. h. $xRy \wedge xRz \Rightarrow yRz$.

Zwei Elemente x und y, die äquivalent sind, werden als $x \sim y$ geschrieben. Die *Äquivalenzklasse* $[x]$ eines Elements x ist die Menge der Objekte, die äquivalent zu x sind.

2.5 Klassifizierung

Unter *Klassifizierung* versteht man das Zusammenfassen von Objekten zu *Klassen* (Gruppen) anhand bestimmter *Merkmale* der Objekte. Ein mathematisches Modell, das die Klassifizierung durchführt, bezeichnet man als *Klassifikator* und die zugehörige Vorgehensweise als *Klassifikationsverfahren*. Unter *Training* (Lernen) versteht man die Anpassung eines Klassifikators an bestimmte Objektmerkmale. Dies erfolgt anhand sogenannter *Trainingsdaten*, bei denen für jedes Objekt seine Klasssenzugehörigkeit bekannt ist. In diesem Fall spricht man auch von *überwachtem Lernen*.

Zwei wesentliche Eigenschaften eines Klassifikators sind sein Bias und seine Varianz (Hastie et al. 2009; Manning et al. 2008). Der *Bias* ist der Fehler ausgehend von falschen Annahmen, die während des Trainings getroffen werden. Er beschreibt die Fähigkeit eines Klassifikators, die Beziehungen in den Trainingsdaten zu modellieren, und hängt somit stark mit der Komplexität des Klassifikators zusammen. Die *Varianz* bezeichnet hingegen die Empfindlichkeit eines Klassifikators gegenüber Änderungen in den Trainingsdaten. Sie ist groß, wenn unterschiedliche Trainingsdaten zu sehr unterschiedlichen Klassifikatoren führen. Die Varianz stellt somit ein Maß für die Inkonsistenz der Entscheidungen dar, unabhängig davon, ob diese richtig oder falsch sind. Klassifikatoren mit hoher Varianz sind anfällig für *Überanpassung* an Trainingsdaten, die zur Modellierung des Rauschens in diesen führt.

Im Allgemeinen führt ein komplexeres Modell zu kleinerem Bias aber größerer Varianz und umgekehrt. Folglich lassen sich nicht beide gleichzeitig minimieren. Das Ziel des Trainings eines Klassifikators besteht daher darin, lediglich wesentliche Eigenschaften der Trainingsdaten zu modellieren um Überanpassung zu vermeiden (*Bias-Varianz-Dilemma*).

Die Beurteilung eines Klassifikators erfolgt an unabhängigen *Testdaten* und wird als *Testen* bezeichnet. Als Gütemaß dient der *Klassifikationsfehler* (*Testfehler*), der den prozentualen Anteil der falsch klassifizierten Objekte in den Testdaten angibt.

Zur Unterteilung in Trainings- und Testdaten wird meistens die k-fache *Kreuzvalidierung* eingesetzt. Dabei erfolgt zuerst die Unterteilung der Trainingsdaten in k disjunkte Teilmengen. Anschließend werden k Testdurchläufe durchgeführt, wobei der Klassifikator immer abwechselnd an $k-1$ Teilmengen trainiert und an der verbleibenden Teilmenge getestet wird. Der resultierende Klassifikationsfehler errechnet sich als Durchschnitt aus den Klassifikationsfehlern der k Einzeldurchläufe.

Ein häufig eingesetztes Klassifikationsverfahren ist der Random Forest (Breiman 2001). Er basiert auf Entscheidungsbäumen und stellt eine Erweiterung des Baggings (Breiman 1996) dar.

2.5.1 Entscheidungsbaum

Ein *Entscheidungsbaum* ist ein einfaches Klassifikationsverfahren, das einem gerichteten Baum zur Beschreibung hierarchisch aufeinanderfolgender Entscheidungen entspricht. Die Klassifizierung erfolgt über die Traversierung des Entscheidungsbaumes, wobei an jedem inneren Knoten die Verzweigung über das zugehörige Merkmal und einen Schwellenwert bestimmt wird.

Die Konstruktion erfolgt anhand rekursiver Unterteilung der Trainingsdaten anhand eines Merkmals und eines Schwellenwerts, die ein bestimmtes Gütemaß optimieren. Die am häufigsten eingesetzten Gütemaße hierfür sind die Gini Impurity (Breiman et al. 1984) und die Entropie (Quinlan 1996). Das Merkmal zusammen mit dem Schwellenwert beschreiben einen inneren Knoten des Entscheidungsbaumes. Das Training wird gestoppt, wenn keine weitere Unterteilung der Trainingsdaten mehr möglich ist. Die Blätter des Entscheidungsbaumes bestimmen schließlich die Klassenzugehörigkeit, können aber auch die Klassenwahrscheinlichkeit enthalten. Die Letztere ergibt sich aus dem Verhältnis der Objekte der Klasse zur Gesamtanzahl der im Blatt enthaltenen Objekte.

2.5.2 Bagging

Die Komplexität eines Entscheidungsbaumes ist durch seine Höhe gegeben. Folglich verringert sich mit steigender Höhe der Bias, aber die Varianz erhöht sich. Um die Varianz gering zu halten, wird häufig die Tiefe der Entscheidungsbäume begrenzt (engl. *pruning*).

Eine alternative Möglichkeit bietet Bootstrap Aggregation – Bagging (Breiman 1996). Hierbei werden mehrere (unbeschnittene) Entscheidungsbäume mit geringem Bias kombiniert, um die Varianz zu reduzieren. Ausgehend von einer Trainingsdatenmenge T der Größe n werden k Teilmengen T_i der Größe m über gleichverteiltes Ziehen von Objekten aus T mit Zurücklegen generiert (engl. *bootstrap sample*). Durch das Ziehen mit Zurücklegen können einige Objekte mehrfach vorkommen. Wenn $m = n$ und n groß ist, enthält eine Teilmenge T_i in etwa $2/3$ (exakt: $1 - 1/e$) unterschiedliche Objekte aus T, die restlichen $1/3$ sind Duplikate. Anschließend werden k Entscheidungsbäume für jede Teilmenge T_i trainiert. Die Klassifikation erhält man schließlich über die Kombination der Klassifikationen der k einzelnen Entscheidungsbäume.

2.5.3 Random Forest

Eine Erweiterung von Bagging mit dem Ziel, die Varianz weiter zu reduzieren, stellt der Random Forest (Breiman 2001) dar. Bagging mit k Entscheidungsbäumen, jeder mit der Varianz σ^2, führt zu der Varianz $\frac{1}{k}\sigma^2$. Wenn die Entscheidungsbäume paarweise eine positive Korrelation ρ aufweisen, führt Bagging zur Varianz (Hastie et al. 2009)

$$\rho\sigma^2 + \frac{1-\rho}{k}\sigma^2. \tag{2.20}$$

Mit steigender Anzahl k verschwindet der zweite Term, aber der erste verbleibt. Das führt dazu, dass der Vorteil der Kombination von mehreren Entscheidungsbäumen durch die Korrelation zwischen Paaren von Entscheidungsbäumen begrenzt wird. Im Random Forest wird daher versucht, die Korrelation zwischen diesen zu verringern, ohne die Varianz zu stark zu erhöhen. Dies wird während der Konstruktion der einzelnen Entscheidungsbäume durch Bagging und eine zufällige Auswahl von m Merkmalen für jeden inneren Knoten realisiert. Die optimale Unterteilung der Trainingsdaten basiert dann ausschließlich auf diesen zufällig ausgewählten Merkmalen.

Ein Random Forest weist folgende Eigenschaften auf:

- *Robust gegenüber Überanpassung*: Die Erhöhung der Anzahl der Entscheidungsbäume führt nicht zur Überanpassung (Breiman 2001). Allerdings können zu hohe Entscheidungsbäume die Varianz erhöhen. Dies beeinflusst die Klassifikation aber meistens unwesentlich, sodass eine Überanpassung selten festgestellt werden konnte (Díaz-Uriarte und Andrés 2006; Hastie et al. 2009).
- *Out-of-Bag (OOB) Fehler*: Aufgrund von Bagging werden etwa 1/3 der Trainingsdaten zum Trainieren nicht verwendet. Diese bilden die sogenannten *Out-of-Bag-Daten* und können als Testdaten zur Bestimmung des Klassifikationsfehlers eingesetzt werden. Dies entspricht weitgehend der Kreuzvalidierung, sodass diese nicht explizit durchgeführt werden muss (Breiman 2001; Breiman 2004).
- *Bedeutung der Merkmale* (engl. *variable importance*): Mittels Out-of-Bag-Daten kann eine Abschätzung hinsichtlich der Bedeutung einzelner Merkmale für die Klassifikation gegeben werden.
- *Parallelisierbarkeit*: Aufgrund der Unabhängigkeit der einzelnen Entscheidungsbäume können diese parallel verarbeitet werden. Das führt zu einer effizienten Klassifizierung.

Des Weiteren sind während des Trainings lediglich zwei relevante Parameter anzupassen: Der erste Parameter ist die Anzahl k der Entscheidungsbäume und der zweite die Anzahl m an zufällig auszuwählenden Merkmalen.

Mit steigendem k verringert sich die Varianz, aber die Trainings- und die Klassifizierungszeit sowie die erforderlichen Ressourcen steigen linear an. Aufgrund der Robustheit gegenüber Überanpassung verbleibt die Effizienz als einziges Kriterium für ein niedriges k. In (Breiman 2004) wird empfohlen $k \geq 1000$ bis hin zu $k = 5000$ zu verwenden, um eine zuverlässige Variable Importance zur erhalten. Allerdings ist zu beachten, dass sich die Klassifikationsgüte ab einem bestimmten k nicht mehr signifikant verbessert.

Ein niedriger Wert für m führt zur reduzierter Korrelation zwischen Paaren von Entscheidungsbäumen. Allerdings erhöht sich hierdurch der Bias leicht, die reduzierte Varianz kompensiert diese Erhöhung aber weitgehend, was im Allgemeinen zu einem besseren Klassifikator führt. Ein Random Forest ist nicht sehr empfindlich bezüglich der Wahl von m, sodass selbst $m = 1$ zu einer guten Klassifikation führen kann (Breiman 2001). In (Breiman 2004) wird daher vorgeschlagen, aus $m \in \{\sqrt{m}/2, \sqrt{m}, 2\sqrt{m}\}$ den Wert, der zum niedrigsten Klassifikationsfehler führt, zu benutzen. Im Allgemeinen ist m problemabhängig und sollte explizit ermittelt werden (Hastie et al. 2009).

Wenn die Anzahl der Merkmale groß, der Anteil der relevanten jedoch gering ist, steigt die Wahrscheinlichkeit einer schwachen Klassifikation für kleine m. Der Grund ist, dass die Wahrscheinlichkeit gering ist, dass das relevante Merkmal während der Konstruktion ausgewählt wird. Mit steigender Anzahl an relevanten Merkmalen ist der Random Forest allerdings robust gegenüber weniger relevante Merkmale (Breiman 2001; Hastie et al. 2009).

Erweiterungen von Random Forest sowie weitere theoretische Grundlagen sind in (Biau und Scornet 2016), (Criminisi und Shotton 2013) sowie (Breiman 2001) zu finden.

2.6 Clusteranalyse

Unter *Clusteranalyse* versteht man Verfahren zur Strukturentdeckung in den Daten. Hierbei erfolgt die Gruppierung von Datenobjekten zu sogenannten *Clustern*, wobei Objekte innerhalb des gleichen Clusters eine höhere Ähnlichkeit zueinander aufweisen als zu den Objekten anderer Cluster.

Hierarchische Clusteranalyse ist eine Familie von Verfahren zur Bestimmung einer Hierarchie von Clustern. Dabei wird zwischen folgenden beiden Vorgehensweisen unterschieden:

- *Agglomerative Clusterverfahren*: Zu Beginn bildet jedes Objekt einen Cluster. Diese werden dann schrittweise zu immer größeren zusammengefasst, bis alle Objekte zu einem Cluster gehören.
- *Divisive Clusterverfahren*: Zu Beginn befinden sich alle Objekte in einem Cluster. Dieses wird dann rekursive in immer kleinere Cluster aufgeteilt, bis jeder Cluster nur noch aus einem Objekt besteht.

Unabhängig von der Vorgehensweise müssen Ähnlichkeitsmaße zur Bestimmung des Abstandes zwischen zwei Objekten (z. B. Jaccard-Distanz (Jaccard 1912), Tanimoto-Distanz (Rogers und Tanimoto 1960)) und zwischen zwei Clustern definiert werden (z. B. Single- oder Complete-Linkage (Murtagh und Contreras 2012)).

3 Stand der Forschung

Zu den Ersten, die sich mit der automatischen Schätzung von Kameraposen in Bildsequenzen beschäftigten, gehörten (Fitzgibbon und Zisserman 1998) und (Koch et al. 1998). Später zeigten Schaffalitzky und Zisserman (2002), dass die automatische Kameraposeschätzung auch ohne Ausnutzung von Zusatzinformation möglich ist. Ihr Ansatz wies allerdings eine hohe Komplexität auf, weswegen er nur für kleinere Bildmengen geeignet war. Mit dem System Photo Tourism gelang es Snavely et al. (2006) erstmals auch größere Bildmengen zu verarbeiten. Das Verfahren zur Poseschätzung aus Photo Tourism wurde unter dem Namen Bundler weiterentwickelt. Es lieferte eine hohe Qualität, skalierte jedoch aufgrund der exzessiven Bildzuordnung zwischen allen möglichen Bildpaaren zusammen mit der inkrementellen Vereinigung unzureichend für große Bildmengen. Darauf folgten Arbeiten mit dem Ziel, immer größere Bildmengen effizient verarbeiten zu können, wobei Bundler häufig noch als Referenz für die erzielbare Qualität herangezogen wird.

In (Snavely et al. 2008) wurde die Problemstellung in Abhängigkeit von der Komplexität der Szene anstatt der Anzahl an Bildern formuliert. Li et al. (2008) ermittelten hingegen über Clusteranalyse repräsentative Bilder, die sie zur Einschränkung relevanter Bildpaare benutzten. Das Hauptziel in (Agarwal et al. 2009) bestand hingegen in einer schnelleren Verarbeitung durch Verteilung der Berechnungen auf einem Computercluster. Frahm et al. (2010) zeigten mit der Erweiterung des Verfahrens von (Li et al. 2008), dass sich eine weitere Effizienzsteigerung durch die Ausnutzung massiver Parallelisierung auf Grafikkarten erzielen lässt.

Aufbauend auf den Arbeiten von (Li et al. 2008) und (Frahm et al. 2010) schafften es Heinly et al. (2015) schließlich, unter anderem durch effiziente Implementierungen auf hochgradig parallelen Rechensystemen, Bildmengen bestehend aus mehreren Millionen Bildern innerhalb weniger Tage zu verarbeiten.

Eine signifikante Effizienzsteigerung ohne Verwendung spezieller Architekturen gelang mit der Einführung einer speziellen Datenstruktur genannt Vocabulary Tree (Sivic und Zisserman 2003), die eine effiziente Reduktion von relevanten Bildpaaren ermöglicht (siehe Abschnitt 3.1.5). Ansätze wie (Agarwal et al. 2009; Klopschitz et al. 2010; Havlena et al. 2013) oder (Schönberger und Frahm 2016) nutzen unter anderem diese Datenstruktur, um eine ausreichende Skalierung für die Verarbeitung von großen Bildmengen zu erzielen.

Zwei wesentliche Eigenschaften eines Verfahrens für die automatische Kameraposeschätzung sind seine Effizienz (Abschnitt 3.1) und seine Robustheit gegenüber komplexen Kamerakonfigurationen. Unter Letzterer ist die Fähigkeit gemeint, auch stark unterschiedliche Bilder miteinander verknüpfen sowie mit kritischen Kamerakonfigurationen umgehen zu können. Hierzu wurde eine Vielzahl robuster Bildzuordnungsverfahren (Abschnitte 2.1.4 und 2.1.5) sowie Merkmalsoperatoren (Abschnitt 2.1.2) entwickelt. Mit Ansätzen zur Detektion kritischer Kamerakonfigurationen

beschäftigt sich Abschnitt 3.2. Unterschiedliche Modellierungen der Beziehungen zwischen den Bildern in komplexen Bildmengen werden in Abschnitt 3.3 beschrieben.

3.1 Effizienzsteigerung für automatische Kameraposeschätzung

Die Effizienz eines Verfahrens zur Kameraposeschätzung wird im Wesentlichen durch die Bildzuordnung und die Bündelausgleichung bestimmt. Letztere kann durch Ausnutzung spezieller Eigenschaften beschleunigt werden (Mitra und Chellappa 2008; Agarwal et al. 2010; Jian et al. 2011; Jeong et al. 2012). Zudem wurden hierarchische Verfahren für die Vereinigung von Bildern (Fitzgibbon und Zisserman 1998; Shum et al. 1999; Farenzena et al. 2009; Mayer 2014) entwickelt, die im Vergleich zu inkrementellen (Snavely et al. 2006; Mayer et al. 2012) parallelisierbar und somit auf entsprechenden Architekturen effizienter sind (vgl. Abschnitt 2.1.8).

Die fehlende Kenntnis der Überlappungen zwischen den Bildern in Bildmengen muss im Rahmen der Kameraposeschätzung ermittelt werden. Einige Verfahren (Schaffalitzky und Zisserman 2002; Snavely et al. 2006; Snavely et al. 2008) führen hierzu eine *vollständige Bildzuordnung* zwischen allen möglichen Bildpaaren durch, was selbst bei kleineren Bildmengen unzureichend skaliert. Wu (2012) beschleunigte hierzu Bildzuordnung über die Parallelisierung auf Grafikkarten. In großen Bildmengen verfügt jedoch die Mehrheit der Bilder über keine Überlappung, sodass deren Bestimmung den Rechenaufwand dominiert. In diesem Fall stellt die Bildzuordnung oft den aufwändigsten Verarbeitungsschritt eines Verfahrens zur Kameraposeschätzung dar.

Unter Bildzuordnung versteht man die Bestimmung korrespondierender Merkmale zwischen den Bildern (vgl. Abschnitt 2.1.2). Die exakte Korrespondenzsuche erfordert, dass für jedes Merkmal in einem Bild, die am besten korrespondierenden Merkmale in jedem anderen Bild ermittelt werden. Geht man von deskriptorbasierten Merkmalen aus, so entspricht dies der Nächsten-Nachbar-Suche im hochdimensionalen Deskriptorraum, gefolgt von der Verfeinerung der Korrespondenzen über die geometrische Verifikation (vgl. Abschnitt 2.1.4). Daraus ergibt sich eine quadratische Komplexität, die zum einen von der Bildanzahl und zum anderen von der Anzahl an Merkmalen pro Bild abhängig ist.

3.1.1 Näherungsverfahren zur Nächsten-Nachbar-Suche

Die Beschleunigung der Nächsten-Nachbar-Suche in niedrigdimensionalen Räumen erfolgt oft durch Unterteilung des Raumes mittels k-d-Baum (Friedman et al. 1977) oder ähnlicher Datenstrukturen (Samet 2006). In höherdimensionalen Deskriptorräumen (bspw. 128 Dimensionen beim SIFT-Deskriptor) skaliert jedoch im Allgemeinen kein Verfahren für die exakte Nächste-Nachbar-Suche. Das heißt, keine Vorgehensweise führt aufgrund des sogenannten *Fluchs der Dimensionalität* (Bellman 1957) zu einer Verbesserung im Vergleich zur linearen Suche. Selbst hochgradig parallelisierte Nächste-Nachbar-Suche wie in (Wu 2012) skaliert unzureichend für größere Bildmengen.

Daher wurden Näherungsverfahren entwickelt, die effizient einen Nachbarn liefern, der allerdings nicht zwangsweise der nächstliegende ist, jedoch in einem begrenzten Abstand von diesem liegt.

Diese basieren auf modifizierten k-d-Bäumen (Beis und Lowe 1997; Silpa-Anan und Hartley 2008; Muja und Lowe 2009), Clusteranalyse (Fukunage und Narendra 1975; Nistér und Schaffalitzky 2006; Leibe et al. 2006) oder Hashing (Indyk und Motwani 1998; Gionis et al. 1999). Alle diese Näherungsverfahren weisen jedoch eine super-lineare Komplexität auf und sind somit für große Bildmengen wenig geeignet.

3.1.2 Kompakte Deskriptoren

Eine weitere Möglichkeit die Nächste-Nachbar-Suche zu beschleunigen besteht in einer kompakteren Repräsentation von Deskriptoren. Die Aussagekraft eines Deskriptors wird dadurch zwar herabgesetzt, aufgrund der kompakteren Repräsentation sind die Vergleiche jedoch schneller durchführbar.

In (Ke und Sukthankar 2004; Mikolajczyk und Schmid 2005a) wurde hierzu die Dimension des SIFT-Deskriptors mittels Hauptkomponentenanalyse reduziert. Allerdings erfordert dies eine vorherige Trainingsphase zur Schätzung der Kovarianzmatrix für die Projektion in den niedrigdimensionalen Deskriptorraum. Deswegen wurde in (Zhao et al. 2010) die Hadamard-Transformation zur Dimensionsreduktion eingesetzt, die bei ähnlicher Qualität keine Trainingsphase erfordert.

Neben reellen existieren zudem binäre Deskriptoren (Heinly et al. 2012) sowie Verfahren zur Quantisierung von reellen auf binäre Deskriptoren. Letztere ermöglichen eine platzsparende Repräsentation und haben zudem den Vorteil, dass der Vergleich sehr schnell über die Bitoperation XOR (exklusives Oder) durchgeführt werden kann. Das Ziel der Verfahren zur Quantisierung besteht in der Distanzerhaltung, d. h. zwei Deskriptoren, die im reellen Raum nah sind, sollten dies auch nach der Quantisierung sein. Dazu wurde beispielsweise in (Strecha et al. 2012) die lineare Diskriminazanalyse als ein überwachtes Lernverfahren eingesetzt. Cheng et al. (2014) formulierten die Korrespondenzsuche als mehrstufiges Hashing (Charikar 2002), was implizit zur Quantisierung von Deskriptoren ohne die Notwendigkeit einer Trainingsphase führt.

3.1.3 Merkmalsreduktion

Die Komplexität der Bildzuordnung ist direkt von der Anzahl der Merkmale im Bild abhängig. Folglich führt die Verringerung ihrer Anzahl zur Reduktion der Komplexität der Nächsten-Nachbar-Suche um einen konstanten Faktor pro Merkmal in einem Bild.

Eine Möglichkeit hierzu besteht darin, die Anzahl der Merkmale über Anpassung der Detektorparameter zu begrenzen (Hartmann et al. 2016). Auf diese Weise werden lediglich Merkmale mit höherem Kontrast, die sich besser zuordnen lassen, extrahiert. Allerdings ist die Bestimmung geeigneter Detektorparameter schwierig und szenenabhängig. Eine zu restriktive Einstellungen kann demnach zum Verlust essenzieller Merkmale führen, weil diese keinen hohen Kontrast aufweisen. In (Wu 2013) werden hingegen vorzugsweise größere Merkmale benutzt, weil diese dazu tendieren, sich besser zuordnen zu lassen.

In (Hartmann et al. 2014) wurde anhand Trainingsdaten gelernt, welche Deskriptoren für die Bildzuordnung geeignet sind. Über eine binäre Klassifikation erfolgte dann die Auswahl

entsprechender Merkmale in den Bildern. Toldo et al. (2015) führten hingegen eine zweistufige Korrespondenzsuche durch. Zuerst wurde über ein Näherungsverfahren (vgl. Abschnitt 3.1.1) eine kleine, konstante Anzahl an Deskriptoren pro Bild verglichen. Diese beschreiben Merkmalen mit dem gröbsten Maßstab, weil dadurch eine bessere Repräsentation des gesamten Bildes gegeben ist. Basierend auf den ermittelten Korrespondenzen wurden die geeignetsten Bildpaare bestimmt und für diese die Zuordnung basierend auf allen Merkmalen, durchgeführt.

Alle diese Ansätze sind mehr oder weniger szenenabhängig. Während die Wahl der Detektorparameter direkt davon abhängig ist, ergibt sich bei der Klassifikation der Merkmale eine indirekte Abhängigkeit durch die Trainingsdatenmenge. Der Ansatz von (Wu 2013) ist zudem ungeeignet, wenn größere Maßstabsunterschiede zwischen den Bildern auftreten.

3.1.4 Bildreduktion

Die Anzahl der Bilder bestimmt die Komplexität der Bildzuordnung. Somit kann durch die Reduktion auf diejenigen Bilder, die für die Szene essenziell sind, die Komplexität signifikant gesenkt werden. Insbesondere bei großen Bildmengen von bekannten Sehenswürdigkeiten aus dem Internet gibt es bestimmte Szenenbereiche, die von ähnlichen Standpunkten aus vielfach aufgenommen wurden. In solchen Fällen führt die Verwendung von weniger Bildern in diesen Bereichen zu keinen nennenswerten Qualitätseinbußen, reduziert aber die Komplexität der Bildzuordnung erheblich. Basierend auf diesen Gedanken entstanden Verfahren, die diese Redundanz zur Effizienzsteigerung ausnutzen.

Li et al. (2008) gruppierten Bilder basierend auf dem GIST-Deskriptor (Oliva und Torralba 2001) in Cluster, sodass jedes Cluster Bilder beinhaltet, die von ähnlichen Standpunkten aus aufgenommen wurden und somit ähnliche Szenenbereiche zeigen. Für jedes Cluster bestimmten sie ein repräsentatives Bild (Iconic Image), das zur Ermittlung inkonsistenter Bilder benutzt wurde. Eine effizientere Umsetzung dieses Verfahrens erfolgte in (Frahm et al. 2010) über Implementierung auf der Grafikkarte.

Havlena et al. (2013) formulierten die Ermittlung repräsentativer Bilder als Suche nach der minimalen, zusammenhängenden, dominierenden Menge der Knoten im ungerichteten Graphen. Die Knoten des Graphen korrespondieren mit den Bildern und eine Kante zwischen zwei Knoten existiert, wenn sich die zugehörigen Bilder wahrscheinlich überlappen. Dabei wird nach einer Teilmenge der Knoten gesucht, sodass alle Knoten des Graphen entweder in dieser Teilmenge enthalten oder zumindest zu einem der Knoten daraus benachbart sind. Zudem wird gefordert, dass der induzierte Teilgraph zusammenhängend ist.

3.1.5 Paarreduktion

In großen Bildmengen besteht meist zwischen der überwiegenden Mehrzahl an Bildern keine Überlappung. Eine Bildzuordnung zwischen allen möglichen Paaren von Bildern verbringt in diesem Fall die meiste Rechenzeit mit Paaren, die über keine Überlappung und damit über keine Korrespondenzen verfügen. Um solche Paare auszuschließen, wurden Verfahren entwickelt, die

ohne eine explizite Bildzuordnung versuchen, Paare, die sich am wahrscheinlichsten überlappen, zu bestimmen.

Kim et al. (2012) modellierten die Beziehungen zwischen den Bildern über einen Graphen (vgl. Abschnitt 3.3) mit dem Ziel, ihn effizient für große Bildmengen zu konstruieren. Dazu wurde zuerst die Bildzuordnung nur für einige Paare durchgeführt, was zu einem dünnbesetzten Graphen führte. Anschließend ermittelten sie fehlende Kanten mittels spektraler Graphenanalyse. Die Bestimmung initialer Paare für die Bildzuordnung stellt einen kritischen Schritt dar. Wenn die Bilder dieser Paare keine ausreichende Überlappung aufweisen, kann dies zu einem nicht zusammenhängenden Graphen führen, dessen Zusammenhangskomponenten über die spektrale Graphenanalyse nicht nachträglich verbunden werden können.

Eine sehr effiziente und häufig eingesetzte Vorgehensweise zur Reduktion von Paaren besteht in der Verwendung des sogenannten *Vocabulary Tree* (Sivic und Zisserman 2003; Nistér und Stewenius 2006). Dieser entspricht einem k-nären Baum, der im Rahmen einer Trainingsphase auf Grundlage einer großen Menge von Deskriptoren mittels Clusteranalyse aufgebaut wird. Dabei unterteilt man die Deskriptormenge rekursiv in k Cluster, bis die gewünschte Baumtiefe L erreicht ist. Die Korrespondenzsuche entspricht dann dem Vergleich mit den Zentren jedes Clusters und der Auswahl des nächsten. Sivic und Zisserman (2003) erweiterten den Vocabulary Tree um eine effizientere Abfrage mittels eines *Inverted File* (Witten et al. 1999), woraus sich auch die Bildähnlichkeit ergibt.

Die Paarreduktion entspricht der Ermittlung einer festgelegten Anzahl an ähnlichsten Bildern für jedes Bild mittels Vocabulary Tree, wobei ähnliche Bilder über einen Schwellenwert bestimmt werden. Allerdings ist eine zuverlässige Bestimmung dieses Schwellenwertes sowie optimaler Parameter k und L schwierig, woraus einige Erweiterungen entstanden sind (Chum et al. 2007; Philbin et al. 2007; Jegou et al. 2008; Chum et al. 2011; Havlena und Schindler 2014). Dennoch konnte der Vocabulary Tree bereits in einer Vielzahl von Verfahren zur Verarbeitung von großen Bildmengen erfolgreich eingesetzt werden (Agarwal et al. 2009; Klopschitz et al. 2010; Havlena et al. 2010; Havlena et al. 2013; Schönberger und Frahm 2016).

Die Reduktion von Paaren in (Schönberger et al. 2015b) basiert auf der Annahme, dass, wenn die gleiche Szene von unterschiedlichen Standpunkte aus aufgenommen wurde, korrespondierende Merkmale ihre Lage und Rotation in erkennbaren Mustern ändern. Über eine binäre Klassifikation mittels Random Forest wurde dann entschieden, welche Paare überlappenden Bildern entsprechen. Aus Effizienzgründen wurden hierfür geschätzte Korrespondenzen basierend auf Histogrammen benutzt. Im Gegensatz dazu verwendeten Schönberger et al. (2015a) exakte Korrespondenzen, was zu einer genaueren Klassifikation führte. Während das erste Verfahren versuchte die Korrespondenzsuche zu beschleunigen, konzentrierte sich das zweite auf die Reduktion der aufwändigen geometrischen Verifikation nach der Korrespondenzsuche.

3.2 Detektion kritischer Kamerakonfigurationen

Degenerierte Kamerakonfigurationen (vgl. Abschnitt 2.1.6) führen im Allgemeinen zu Problemen während der Kameraposeschätzung, weswegen diese detektiert und entsprechend behandelt werden müssen. Während die Strukturdegeneration bei kalibrierten Kameras unproblematisch ist, führt

die Bewegungsdegeneration immer zu fehlerhafter Geometrieschätzung (vgl. Abschnitt 2.1.6). Das liegt daran, dass die Bilder ausschließlich über eine Homographie in Beziehung stehen und die 3D-Rekonstruktion undefiniert ist. Diese Problematik wurde zuerst in (Kanatani 1996) im Zusammenhang mit der Bestimmung der Fundamentalmatrix angesprochen und weiter in (Torr et al. 1998) und (Torr et al. 1999) thematisiert.

3.2.1 Modellauswahl

In (Torr 1997) wurde ein robustes Verfahren zur Modellauswahl namens Geometric Robust Information Criterion (GRIC) vorgestellt. Es beruht darauf, dass im Falle von kritischen Kamerakonfigurationen zwischen zwei Aufnahmen ihre geometrische Beziehung besser durch eine Homographie als eine Fundamentalmatrix beschreibbar ist und GRIC für diese einen besseren Wert liefert. Es werden somit die Homographie und die Fundamentalmatrix geschätzt und die zugehörigen GRIC-Werte berechnet. Falls die Homographie einen besseren GRIC-Wert liefert, handelt es sich um eine kritische Konfiguration. Diese Vorgehensweise wurde bereits mehrfach eingesetzt, um ungeeignete Kamerakonfigurationen zu detektieren (Gherardi et al. 2010; Toldo et al. 2015; Schönberger und Frahm 2016).

Eine robustere Bestimmung der Fundamentalmatrix selbst dann, wenn die Mehrheit der Korrespondenzen koplanar ist, d. h. sich auf einer sogenannten dominanten Ebene befindet, ermöglicht DEGENSAC (Chum et al. 2005). Es basiert auf RANSAC (siehe Abschnitt 2.1.4), wobei in jeder Iteration die Homographie bzw. die Fundamentalmatrix mit der größten Unterstützerzahl bestimmt wird. Im Falle der Homographie wird die zugehörige Fundamentalmatrix mittels des Plane-and-Parallax-Algorithmus (Irani und Anandan 1996) ermittelt.

Ein Nachteil der bisher beschriebenen Verfahren ist, dass Wissen über die Degeneration bzw. das zugrunde liegende Modell erforderlich ist. Im Gegensatz dazu erfordert RANSAC for (Quasi-)Degenerate Data (QDEGSAC) (Frahm und Pollefeys 2006) kein spezifisches Wissen. Es basiert auf einer iterativen Anwendung von RANSAC unter Verwendung von immer weniger Nebenbedingungen, um das geeignetste Modell zu ermitteln. Korrespondenzen, die durch degenerierte Modelle beschreibbar sind, werden herausgefiltert und das korrekte Modell mit den verbleibenden bestimmt. Somit ist dieses selbst beim Vorliegen einer geringen Anzahl gültiger Korrespondenzen bestimmbar.

3.2.2 Konjugierte Rotation

Wenn die Kamera zwischen den Aufnahmen nicht verschoben wurde, stehen die betroffenen Bilder über eine unendliche Homographie $\boldsymbol{H}_\infty$ in Beziehung (vgl. Abschnitt 2.1.6). Falls die Kalibriermatrizen der Kameras identisch sind, d. h. $\boldsymbol{K} = \boldsymbol{K}_1 = \boldsymbol{K}_2$, so entspricht

$$\boldsymbol{H}_\infty = \boldsymbol{K}\boldsymbol{R}\boldsymbol{K}^{-1} \tag{3.1}$$

einer *konjugierten Rotation*, wodurch $\boldsymbol{H}_\infty$ die gleichen Eigenwerte wie die Rotationsmatrix $\boldsymbol{R}$ (bis auf einen beliebigen Skalierungsfaktor) besitzt[1]. Somit bestimmen ihre komplexen Eigenwerte

[1] Eine Rotationsmatrix $\boldsymbol{R}$ besitzt die Eigenwerte $(1, e^{i\theta}, e^{-i\theta})$ mit dem Rotationswinkel θ. Der Eigenvektor, der mit dem Eigenwert 1 korrespondiert, entspricht der Rotationsachse.

den Rotationswinkel θ um den die Kamera rotiert ist und der Eigenvektor, der mit dem reellen Eigenwert korrespondiert, den Fluchtpunkt der Rotationsachse (Hartley und Zisserman 2004).

Aus diesem Zusammenhang ergibt sich eine Vorgehensweise, reine Rotation zwischen den Aufnahmen zu erkennen. Dazu erfolgt zuerst die Schätzung der Homographie zwischen den Bildern. Im Anschluss wird überprüft, ob diese einer konjugierten Rotation entspricht. Wie in (Torr et al. 1999) jedoch festgestellt wurde, existieren diesbezüglich folgende Probleme:

- Wenn sich zwischen den Aufnahmen die Kameraparameter ändern (beispielsweise aufgrund von Autofokus), entspricht die Homographie nicht mehr einer konjugierten Rotation.
- Eine Homographie induziert durch eine reale Ebene kann auch einer konjugierten Rotation entsprechen, wenn die Kamera eine ebene Bewegung durchführt. In diesem Fall erfolgt die Translation parallel zu der Weltebene und die Rotationsachse ist parallel zur Normalen der Ebene. Dadurch erfolgt die Translation $\boldsymbol{t}$ orthogonal zur Ebenennormalen $\boldsymbol{n}$, sodass aufgrund von $\boldsymbol{t}\boldsymbol{n}^T = 0$ der Term $\boldsymbol{t}\boldsymbol{n}^T/d$ aus der Gleichung (2.13) 0 wird. Somit ist die Homographie induziert durch die reale Ebene konjugiert zu der planaren euklidischen Transformation und ist ebenfalls eine konjugierte Rotation. Ein Beispiel für eine derartige Aufnahmekonfiguration ist eine Kamera, die auf einem Fahrzeug montiert ist und die Bodenebene betrachtet.

In (Kähler und Denzler 2006) wurde die Homographie zwischen den Bildern geschätzt und daraus über die Gleichung (2.13) der Term $\left(\boldsymbol{R} - \frac{\boldsymbol{t}\boldsymbol{n}^T}{d}\right)$ berechnet. Wurde die Kamera nur rotiert, so entspricht dieser Term genau der Rotationsmatrix $\boldsymbol{R}$. Über Singulärwerte wurde dann geschätzt, wie weit der Term von der Rotationsmatrix abweicht und anhand eines Schwellwerts entschieden, ob eine Translation stattgefunden hat.

3.2.3 Gütemaße für kritische Kamerakonfigurationen

Einen etwas anderen Weg als im vorherigen Abschnitt verfolgen Verfahren zur Abschätzung der Qualität eines Bildpaares hinsichtlich Geometriebestimmung (Nistér 2000). Insbesondere bei der Poseschätzung in Videosequenzen ist es essenziell, ausschließlich Geometrie zwischen Paaren mit ausreichender Basis zu bestimmen. Mit steigender Basislänge erhöhen sich jedoch Bildverzerrungen und es kann zu Verdeckungen kommen, die zum Verlust von Merkmalskorrespondenzen führen. Somit müssen Paare, genannt Keyframes, bestimmt werden, die ausreichende Basislänge und Überlappung aufweisen. Außer für Bildsequenzen ist es auch in inkrementellen Verfahren zu Kameraposeschätzung wichtig, ein zuverlässiges Bildpaar zu bestimmen, das als initialer Basisblock dient (vgl. Abschnitt 2.1.3).

Um Keyframes zu ermitteln, wurden Gütemaße für unkalibrierte Kameraposeschätzung basierend auf Modellauswahl entwickelt (vgl. Abschnitt 3.2.1). Farenzena et al. (2008) benutzten hierzu GRIC und Gibson et al. (2002) den Rückprojektionsfehler für die Homographie sowie den Median des Abstands von der Epipolarlinie für die Fundamentalmatrix. Um eine ausreichende Überlappung sicherzustellen, wurde noch die Anzahl der Korrespondenzen in die Gütemaße

miteinbezogen. Stattdessen erfolgte in (Thormählen et al. 2004) die Schätzung der Unsicherheiten der rekonstruierten 3D-Punkten anhand der Kovarianzmatrizen. Dies erfordert jedoch eine Bündelausgleichung, was dieses Verfahren deutlich aufwändiger als die anderen macht.

Nachteile der bisher vorgestellten Verfahren bestehen zum einen in der Notwendigkeit der Schätzung von zwei Modellen und zum anderen in der fehlenden Fähigkeit, mit ebenen Strukturen umzugehen. Im Gegensatz dazu wird in (Beder und Steffen 2006) von kalibrierten Kameras ausgegangen und es werden die Unsicherheiten der rekonstruierten 3D-Punkte anhand ihrer Kovarianzmatrix bestimmt. Seien λ_1, λ_2 und λ_3 die Eigenwerte der Kovarianzmatrix eines 3D-Punktes $\boldsymbol{X}$ mit $\lambda_1 \geq \lambda_2 \geq \lambda_3$. Die Unsicherheit von $\boldsymbol{X}$ ist dann durch die Rundheit

$$\mathrm{R}(\boldsymbol{X}) = \sqrt{\frac{\lambda_3}{\lambda_1}} \tag{3.2}$$

des zugehörigen Fehlerellipsoids gegeben. $\mathrm{R}(\boldsymbol{X})$ liegt zwischen 0 und 1 und hängt nur von der relativen Geometrie der beiden Kameras und den Merkmalspositionen ab. Wenn die Kamerazentren identisch und die Merkmalspositionen korrekt sind, ist die Rundheit gleich Null. Die mittlere Rundheit über alle 3D-Punkte wird in (Beder und Steffen 2006) als Gütemaß für die Eignung der Kamerakonfiguration hinsichtlich 3D-Rekonstruktion verwendet.

3.3 Modellierung von Bildverknüpfungen

Die Beschreibung von Beziehungen zwischen den Bildern in komplexen Bildmengen erfolgt im Allgemeinen über einen gewichteten Graphen, bei dem Knoten den Bildern entsprechen und eine Kante zwischen zwei Knoten existiert, wenn die korrespondierenden Bilder in der geforderten geometrischen Beziehung zueinander stehen. Die Kantengewichte dienen zur Beschreibung der Güte von Beziehungen und entsprechen beispielsweise der Anzahl an Korrespondenzen (Schaffalitzky und Zisserman 2002; Li et al. 2008; Jiang et al. 2013; Toldo et al. 2015) oder der Unsicherheit der geschätzten Pose (Snavely et al. 2008). Daneben finden auch Hypergraphen[2] Verwendung (Ni und Dellaert 2012), die jedoch deutlich komplexer umzusetzen sind.

Häufig werden ungerichtete Graphen verwendet (Schaffalitzky und Zisserman 2002; Steele und Egbert 2006; Zach et al. 2008; Philbin und Zisserman 2008; Zach et al. 2010; Moulon et al. 2013; Toldo et al. 2015), es existieren aber auch Ansätze basierend auf gerichteten Graphen (Snavely et al. 2008; Irschara et al. 2011; Wefelscheid 2013; Wilson und Snavely 2014). Letztere ermöglichen eine genauere Modellierung, erfordern in der Regel jedoch einen höheren Aufwand zur Ermittlung asymmetrischer Beziehungen.

Verfahren basierend auf ungerichteten Graphen verwenden oft minimale bzw. maximale Spannbäume, um die Verarbeitungsreihenfolge der Bilder festzulegen (Schaffalitzky und Zisserman 2002; Steele und Egbert 2006; Zach et al. 2008; Klopschitz et al. 2010; Zach et al. 2010; Toldo et al. 2015). Neben diesen wird eine Reihe anderer graphentheoretischer Konzepte wie normierte Graphenschnitte (Li et al. 2008), minimale dominierende Knotenmenge (Havlena

[2]Ein *Hypergraph* ist eine Verallgemeinerung des Graphen, bei dem eine Kante, genannt *Hyperkante*, mehr als zwei Knoten miteinander verbinden kann.

et al. 2010), spektrale Graphenanalyse (Heath et al. 2010; Kim et al. 2012) oder Betweenes Centrality (Wefelscheid 2013) zur Bestimmung geeigneter Bildverknüpfungen benutzt.

Spannbäume werden auch eingesetzt, um Beziehungen höherer Ordnung wie Bildtriplets im Graphen zu ermitteln (Klopschitz et al. 2010; Toldo et al. 2015). Um diese Beziehungen besser auf Grundlage von Graphen beschreiben zu können, werden weitere Graphen konstruiert, bei denen Knoten den Bildtriplets entsprechen und Kanten zwischen Knoten existieren, wenn die korrespondierenden Triplets gemeinsame Bilder aufweisen (Klopschitz et al. 2010; Jiang et al. 2013; Wefelscheid 2013).

In (Heath et al. 2010) erfolgt die Verknüpfung von Bildern über gemeinsame Bildregionen an Stelle von Merkmalen. Die Modellierung basiert auf einem ungerichteten Graphen, genannt Image Web, bei dem ein Knoten einem Bereich eines Bildes mit verbundenen Bildpunkten entspricht. Die Bereiche werden auf Grundlage von ermittelten Korrespondenzen und der Vereinigung ihrer elliptischen Merkmale (Forssén 2007) gebildet. Eine Kante zwischen zwei Knoten existiert dann, wenn die Bildbereiche der korrespondierenden Bilder unter affiner Transformation ähnlich sind. Das Ziel besteht hierbei nicht in der Kameraposeschätzung, sondern im Herstellen von Beziehungen zwischen Bildern, selbst wenn sich diese visuell deutlich unterscheiden.

4 Bildverknüpfung

Dieses Kapitel befasst sich mit den theoretischen Grundlagen der automatischen Verknüpfungsbestimmung zwischen Bildern. Unter *Bildverknüpfung* versteht man hierbei das Wissen über die *Überlappung*, d. h. die Abbildung gleicher Teilbereiche einer Szene, zwischen den Bildern. Basierend darauf erfolgt die Schätzung der Zwei- und Dreibildgeometrie von Paaren und Triplets mittels *robuster Kameraposeschätzung* (Abschnitt 5.2). Über die *hierarchische Vereinigung* werden schließlich relative Kameraposen der Bilder einer Bildmenge geschätzt (Abschnitt 5.7). Daraus ergeben sich neben effizienter automatischer Bestimmung folgende *Anforderungen* an die Bildverknüpfung:

(A_1) *Minimierung geometrischer Verifikation*: Die Verarbeitung von Bildern mit großen Bildverzerrungen erfordert eine robuste Kameraposeschätzung mit komplexen und somit zeitaufwändigen Operationen. Um eine effiziente Bildverknüpfung zu erzielen, sind diese auf ein Minimum zu reduzieren.

(A_2) *Bildverknüpfung über Triplets*: Um eine höhere Genauigkeit sowie Robustheit zu erreichen erfolgt die Verknüpfung der Bilder über Triplets. Dies muss während der Bestimmung von Bildverknüpfungen berücksichtigt werden.

(A_3) *Verknüpfung der Triplets über Paare*: Für eine robuste Übertragung des Maßstab zwischen den Triplets werden während der hierarchischen Vereinigung Paare benutzt. Somit lassen sich zwei Triplets nur dann miteinander verknüpfen, wenn sie über zwei gleiche Bilder verfügen. Diese Nebenbedingung ist während der Verknüpfungsbestimmung zu beachten.

(A_4) *Vermeidung von Paaren und Triplets mit kritischen Kamerakonfigurationen*: Geht man von einer robusten Kameraposeschätzung aus, so entstehen kritische Kamerakonfigurationen vor allem aufgrund unzureichendem Verhältnis zwischen der Kamerabasis und der Entfernung der Szene (vgl. Abschnitt 2.1.6). Die Verwendung von Paaren und Triplets mit solchen Konfigurationen kann zu einer fehlerhaften Kameraposeschätzung führen. Folglich müssen diese detektiert und aus der Bildverknüpfung ausgeschlossen werden.

(A_5) *Vollständigkeit*: Es wird von keiner großen Redundanz in der Bildmenge ausgegangen. Daher sind möglichst alle enthaltenen Bilder miteinander zu verknüpfen.

Die Anforderung hinsichtlich der Minimierung geometrischer Verifikationen wird über eine effiziente Schätzung der Bildähnlichkeiten (Abschnitt 4.1) erfüllt. Diese ermöglicht die Reduktion der Anzahl von Paaren und Triplets für welche die robuste Kameraposeschätzung durchzuführen ist. Die Einbeziehung der Anforderung (A_4) erfolgt über die Unterteilung von Paaren und Triplets in unterschiedliche Teilmengen gemäß Abschnitt 4.2. Modellierung von Bildverknüpfungen unter Berücksichtigung der Anforderungen (A_2) und (A_3) ist das Thema der Abschnitte 4.3 bis 4.6. Basierend hierauf wird die Konstruktion einer minimalen Bildverknüpfung im Abschnitt 4.7

beschrieben. Schließlich werden weitere Verknüpfungen zwischen den Bildern hergestellt, um die Robustheit und die Genauigkeit der Kameraposeschätzung zu erhöhen (Abschnitt 4.8).

4.1 Schätzung der Bildähnlichkeiten

Die Grundlage für die Bildverknüpfung stellen paarweise Beziehungen zwischen den Bildern dar. Aufgrund der quadratischen Komplexität ist ihre exakte Bestimmung für eine größere Anzahl von Bildern extrem aufwändig. Selbst Verfahren wie in (Wu 2013), die die Kameraposeschätzung mittels Implementierung auf der Grafikkarte beschleunigen, skalieren für große Bildmengen unzureichend.

Stattdessen wird häufig der sogenannte Vocabulary Tree (Nistér und Stewenius 2006) eingesetzt, um die kombinatorische Komplexität zu reduzieren (vgl. Abschnitt 3.1.5). Dabei handelt es sich um eine spezielle Datenstruktur, die zwar sehr gut skaliert, jedoch eine explizite Trainingsphase erfordert. Außerdem ist ihre Genauigkeit und Effizienz stark von der Parameterwahl abhängig. Des Weiteren können auch wiederholte Strukturen die Genauigkeit beliebig verschlechtern (Irschara et al. 2011).

Die Anzahl der zu betrachtenden Paare lässt sich über die Ähnlichkeiten zwischen den Bildern reduzieren. Das heißt Bilder, die eine geringe Ähnlichkeit aufweisen, überlappen sich wahrscheinlich auch unzureichend für eine erfolgreiche Geometrieschätzung. Einen guten, anwendungsspezifischen Indikator für die Ähnlichkeit stellt die Anzahl der Korrespondenzen zwischen den Bildern dar. Ihre Aussagekraft ist allerdings stark von der eingesetzten Zuordnungsmethode abhängig. Komplexe Zuordnungsmethoden sind genauer aber auch rechenaufwändiger. Aus Effizienzgründen werden daher in dieser Arbeit lediglich Merkmalsdeskriptoren miteinander verglichen. Anhand der darauf basierenden Korrespondenzanzahl wird dann auf die Bildähnlichkeit geschlossen.

Im Rahmen der robusten Kameraposeschätzung werden SIFT-Merkmale zur Extraktion von Bildausschnitten benutzt (vgl. Abschnitt 2.1.2). Folglich erfolgt die Schätzung der Bildähnlichkeiten hier auf der Grundlage von SIFT-Deskriptoren. Ein vollständiger Deskriptorvergleich im euklidischen Deskriptorraum skaliert jedoch aufgrund der quadratischen Komplexität und hochdimensionalen SIFT-Deskriptoren unzureichend (vgl. Abschnitt 3.1). Aus diesem Grund werden in dieser Forschungsarbeit reelle SIFT-Deskriptoren auf binäre quantisiert, um einen effizienten Vergleich mittels Hamming-Distanz

$$\mathrm{d_H}(\boldsymbol{u},\boldsymbol{v}) = \sum_{i=1}^{128} \left[(u_i \circ v_i) := \begin{cases} u_i \neq v_i, & 1 \\ u_i = v_i, & 0 \end{cases} \right] \tag{4.1}$$

zu ermöglichen. Hierbei entsprechen $\boldsymbol{u}$ und $\boldsymbol{v}$ binären SIFT-Deskriptoren mit ihren Elementen u_i und v_i. Der XOR-Bitoperation $\mathrm{d_H}$ zwischen $\boldsymbol{u}$ und $\boldsymbol{v}$ folgt eine Bitzählung, um die Anzahl an unterschiedlichen Bitstellen zu ermitteln. Moderne Prozessoren bieten, unter anderem über ihre Streaming SIMD Extensions (SSE) zur Beschleunigung von Bitoperationen, für beide Operationen eine direkte Unterstützung, was zu sehr effizienten Vergleichen zwischen Deskriptoren $\boldsymbol{u}$ und $\boldsymbol{v}$ führt. Ein weiterer Vorteil von binären Deskriptoren liegt in ihrer Kompaktheit und somit einer sehr speichereffizienten Repräsentation.

4.1.1 Quantisierung von Deskriptoren

Die Quantisierung von d-dimensionalen Deskriptoren führt zu deren Einbettung vom reellen Raum $\mathbb{R}^d$ in den *Hamming-Raum* $\mathbb{H}^d = \{0,1\}^d$. Existierende Verfahren konzentrieren sich auf die Distanzerhaltung, d. h. zwei Deskriptoren, die in $\mathbb{R}$ nah zueinander liegen, sollten ebenfalls in $\mathbb{H}$ nah sein (vgl. Abschnitt 3.1.2). Zur Bestimmung der Bildähnlichkeiten ist allerdings eine Bildordnung ausreichend, die lediglich relative Ähnlichkeiten erfordert. Somit kann eine vereinfachte Einbettung eingesetzt werden, solange die Näherungsfehler gering und gleich verteilt sind.

Ein d-dimensionaler reeller Vektorraum $\mathbb{R}^d$ kann durch d unabhängige (affine) *Hyperebenen* unterteilt werden, wobei jede die Kodimension 1 in $\mathbb{R}^d$ besitzt. Jede Hyperebene geht durch den Schnittpunkt $\boldsymbol{p} \in \mathbb{R}^d$ und unterteilt $\mathbb{R}^d$ in zwei *Halbräume*, genannt positiver und negativer Halbraum. Schnitte der d gegenseitig orthogonalen Halbräumen bestimmen 2^d *Orthante*, die eine Verallgemeinerung der Quadranten in höher dimensionalen Räumen sind (Grünbaum 2003). Jeder Orthant ist durch eine Sequenz von d Plus oder Minus Vorzeichen bestimmt, wobei das i-te Vorzeichen angibt, ob der Orthant im positiven oder negativen Halbraum der i-ten Hyperebene liegt. Somit kann ein Orthant in $\mathbb{R}^d$ durch einen *Bitvektor*[1] der Länge d beschrieben werden.

Ein 128-dimensionaler SIFT-Deskriptor zeigt auf einen der 2^{128} Orthanten. Folglich kann er über einen Bitvektor der Länge 128 repräsentiert werden, der zum Orthant, auf den er zeigt, korrespondiert. Zur Einbettung müssen 128 Hyperebenen bestimmt werden. Die Werte des normierten SIFT-Deskriptors liegen im Bereich zwischen 0 und 1. Daher ist der Ursprung als Schnittpunkt $\boldsymbol{p} = \boldsymbol{0}$ für die Hyperebenen ungeeignet, was eine explizite Bestimmung von $\boldsymbol{p}$ erforderlich macht. Diese erfolgt für die i-te Koordinate des Schnittpunktes $\boldsymbol{p}$ über den Median der Deskriptorwerte der Dimension i.

4.1.2 Ähnlichkeitsschätzung

Der Vergleich von eingebetteten Deskriptoren entspricht der Bestimmung der Anzahl an korrespondierenden Halbräumen über die Hamming-Distanz. Die resultierenden Korrespondenzen werden anschließend mittels des Verhältnistests von Lowe (2004) verfeinert. Basierend darauf erfolgt die Definition der Ähnlichkeit zwischen zwei Bildern i und j über den Jaccard-Index (Jaccard 1912)

$$\mathrm{J}(i,j) = \frac{|M_i \cap M_j|}{|M_i \cup M_j|}, \tag{4.2}$$

wobei M_i und M_j den Merkmalsmengen und $M_i \cap M_j$ der Korrespondenzmenge entspricht. $\mathrm{J}(i,j)$ kann als die Wahrscheinlichkeit, dass beide Bilder ein zufällig ausgewähltes Merkmal gemeinsam haben, betrachtet werden (Liben-Nowell und Kleinberg 2003). Der Wertebereich von J liegt zwischen 0 und 1, wobei 1 die maximale Ähnlichkeit bedeutet.

Wie im Abschnitt 6.2 gezeigt, ermöglicht J eine ausreichende Abschätzung der Bildähnlichkeiten zur Reduktion der Anzahl an nötigen geometrischen Verifikationen mittels robusten Kameraposeschätzung. Die Genauigkeit wird durch den Vergleich der eingebetteten Deskriptoren zwar zuerst

[1] Ein *Bitvektor* ist eine Folge von Bits mit Wertebereich $\{0,1\}$, der effiziente logische Operationen ermöglicht. Die *Länge* des Bitvektors entspricht der Anzahl der enthaltenen Bits.

reduziert. Aufgrund des effizienten Vergleichs können allerdings alle Paare verglichen werden, was wiederum zur Erhöhung der Genauigkeit führt. Zudem ist selbst die korrekte Ähnlichkeit zwischen den Bildern allein nur bedingt nutzbar, weil auch andere Faktoren (z. B. Überlappungsbereich, Verteilung der Merkmale) für die Poseschätzung eine wichtige Rolle spielen. Somit eignen sich die geschätzten Bildähnlichkeiten nach Gleichung (4.2) für eine effiziente und sinnvolle Minimierung der Anzahl an geometrischen Verifikationen.

4.2 Unterteilung von Paaren und Triplets

Für eine fehlerfreie Kameraposeschätzung müssen ungeeignete Paare und Triplets herausgefiltert werden. Diese lassen sich entsprechend Abb. 4.1 in verschiedene Teilmengen unterteilen. Die Unterteilung erfolgt für Paare und Triplets analog, weswegen diese im weiteren Verlauf zusammenfassend als Elemente bezeichnet werden. Eine Elementmenge $\mathcal{E}$ korrespondiert demnach mit der Paarmenge $\mathcal{P}$ bzw. der Tripletmenge $\mathcal{T}$ und ein Element $E \in \mathcal{E}$ mit einem Paar $P \in \mathcal{P}$ bzw. Triplet $T \in \mathcal{T}$.

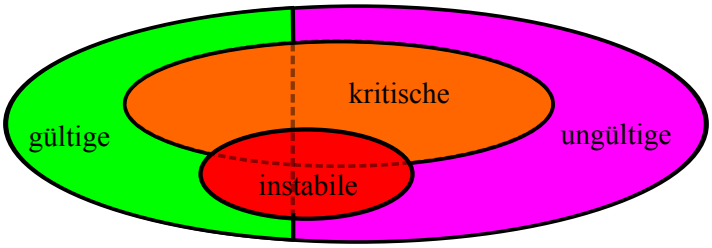


Abbildung 4.1: Unterteilung von verifizierten Paaren und Triplets

Die Elementmenge $\mathcal{E}$ setzt sich im Allgemeinen aus der Teilmenge der gültigen $\mathcal{E}^+$, der ungültigen $\mathcal{E}^-$ sowie unbestimmten Elementen $\mathcal{E} \setminus (\mathcal{E}^+ \cup \mathcal{E}^-)$ zusammen. Ein Element $E \in \mathcal{E}^+$ wird als *gültig* bezeichnet, falls seine Geometrie mittels robuster Kameraposeschätzung bestimmbar ist. Ansonsten bezeichnet man es als *ungültig* und ordnet es der Teilmenge der ungültigen Elemente zu, d. h. $E \in \mathcal{E}^-$. Elemente, die noch nicht mittels robuster Kameraposeschätzung *verifiziert* wurden (vgl. Abschnitt 5.2), gehören zur Teilmenge der unbestimmten Elemente.

Elemente $E \in \mathcal{E}^\times$ weisen kritische Kamerakonfigurationen auf und werden *instabil* genannt (vgl. Abschnitt 2.1.6). Es wird davon ausgegangen, dass eine zuverlässige Geometriebestimmung für diese Elemente nicht möglich ist. Insbesondere bei komplexen Kamerakonfigurationen ist die Detektion von instabilen Elementen schwierig. Aus diesen Grund erfolgt eine weitere Unterteilung in *kritische* Elemente $E \in \mathcal{E}^\sim$, für die keine zuverlässige Aussage hinsichtlich ihrer Stabilität gemacht werden kann.

Letztendlich können auch ungültige Elemente kritisch oder instabil sein (z. B. zwei um 180° gedrehte Aufnahmen ohne Verschiebung dazwischen). Für eine Bildverknüpfung werden demnach ausschließlich *stabile* Elemente $E \in \mathcal{E}^* = \mathcal{E}^+ \setminus (\mathcal{E}^\sim \cup \mathcal{E}^\times)$ verwendet.

4.3 Bildgraph

Direkte Beziehungen zwischen Bildern lassen sich über einen gewichteten, ungerichteten Graphen $G_I = (V_I, E_I)$ modellieren. Knoten entsprechen den Bildern und Kanten den Paaren. Eine Kante zwischen zwei Knoten existiert, falls korrespondierende Bilder einen gemeinsamen Überlappungsbereich aufweisen.

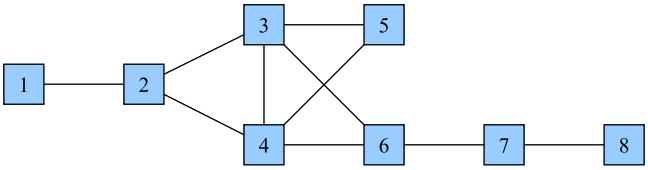


Abbildung 4.2: Bildgraph ohne Kantengewichte

G_I wird als *Bildgraph* (Hartmann et al. 2016), *Epipolargraph* (Klopschitz et al. 2010; Irschara et al. 2011; Wilson und Snavely 2014; Toldo et al. 2015), *Match Graph* (Philbin und Zisserman 2008; Agarwal et al. 2009; Kim et al. 2012), *Viewing Graph* (Levi und Werman 2003; Sweeney et al. 2015) oder auch *Kameranachbarschaftsgraph* (Steele und Egbert 2006) bezeichnet. Aufgrund der Korrespondenz zwischen Knoten und Bildern wird im weiteren Verlauf die Bezeichnung Bildgraph verwendet. Ein Beispiel für einen Bildgraphen, der Beziehungen zwischen acht Bildern modelliert, ist in Abb. 4.2 gegeben. Bilder 3, 4 und 6 besitzen beispielsweise einen gemeinsamen Überlappungsbereich und könnten ein Triplet bilden, während davon ausgegangen wird, dass Bilder 5 und 8 über keinen gemeinsamen Bereich verfügen.

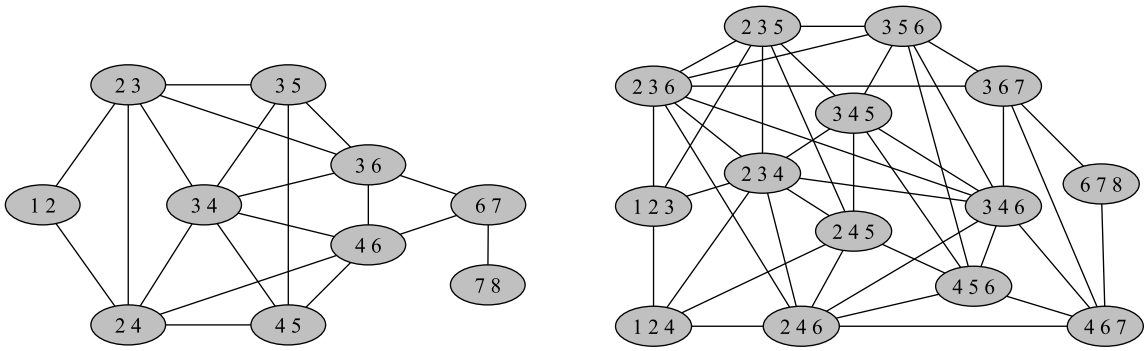
Die Kantengewichtsfunktion basiert auf der Korrespondenzanzahl und ist somit von der Überlappung zwischen den Bildern abhängig. Diese steht wiederum in direkter Beziehung zur Bildähnlichkeit, d. h. je größer die Überlappung zwischen zwei Bildern ist, desto ähnlicher sollten diese sein. Folglich lässt sich die Kantengewichtsfunktion allgemein als ein Ähnlichkeitsmaß zwischen den Bildern korrespondierend mit den inzidenten Knoten formulieren. Basierend darauf erfolgt hier die Definition der Gewichtsfunktion für eine Kante durch den Jaccard-Index J aus der Gleichung (4.2) zwischen den Bildern korrespondierend mit den inzidenten Knoten.

4.4 Paargraph

Der Bildgraph ermöglicht eine intuitive Modellierung der Bildverknüpfung basierend auf den Paaren, da er genau die paarweisen Beziehungen zwischen den Bildern beschreibt. Im Falle von Triplets fehlt dem Bildgraphen jedoch die Möglichkeit, diese Beziehungen höherer Ordnung zu beschreiben (vgl. Abschnitt 4.4.1). Aufgrund der Anforderung (A_3) kommt noch hinzu, dass Triplets über Paare verknüpft werden, was mittels Bildgraphen nicht modellierbar ist. Daher wird ein auf dem Kantengraphen des Bildgraphen basierendes Modell eingeführt, das in der Lage ist, Bildverknüpfungen entsprechend Anforderungen (A_2) und (A_3) vollständig zu beschreiben und zudem die Verwendung konventioneller Graphenalgorithmen erlaubt.

Der Kantengraph $G_P = L(G_I) = (V_P, E_P)$ eines Bildgraphen $G_I = (V_I, E_I)$ besitzt Knoten, die den Kanten von G_I entsprechen. Knoten von G_P korrespondieren somit mit den Paaren, was

zu der Bezeichnung *Paargraph* führt. Zwei Knoten sind benachbart, wenn zugehörige Paare ein gemeinsames Bild beinhalten. Daraus ergibt sich, dass jede Kante in G_P mit einem Triplet korrespondiert. Eine Traversierung durch den Paargraphen ergibt einen Pfad, der implizit einer Verknüpfung von Triplets über Paare und somit einer Bildverknüpfung nach den Anforderungen (A_2) und (A_3) entspricht.



(a) Paargraph (b) Tripletgraph

Abbildung 4.3: Paar- und Tripletgraph als Kantengraphen des Bildgraphen aus Abb. 4.2. Redundante Knoten im Tripletgraphen wurden zusammengefasst.

Die Bestimmung der Geometrie eines Triplets (s_1, m, s_2) mittels robuster Kameraposeschätzung basiert auf den Posen der Slavebilder s_1 und s_2 relativ zum Masterbild m (vgl. Abschnitt 2.1.5). Übertragen auf den Paargraphen korrespondieren die Paare (m, s_1) und (m, s_2) mit den Paarknoten und die inzidente Kante mit dem Triplet (s_1, m, s_2) bzw. dem Paar (s_1, s_2). Das gemeinsame Bild der Paarknoten entspricht dem Masterbild m. Der Paargraph impliziert somit auch ein geeignetes Masterbild.

Die Existenz eines Dreiecks im Paargraphen bezüglich eines Triplets bedeutet, dass jedes Bild als Masterbild geeignet ist. Beispielsweise gehört im Dreieck $(2,3) - (3,4) - (2,4)$ im Paargraphen aus Abb. 4.3 jede Kante zum selben Triplet, führt aber zu unterschiedlichen Masterbildern. Folglich kommt jedes der Bilder 2, 3 oder 4 als Masterbild in Frage. Im Gegensatz dazu sind die Paarknoten $(1,2)$ und $(2,3)$ zwar verbunden, es fehlt jedoch der Paarknoten $(1,3)$. Das bedeutet, dass die Bilder 1 und 3 eine ungeeignete Kamerakonfiguration aufweisen und nicht als Paar zur Tripletkonstruktion verwendet werden können. Es existiert jedoch eine ausreichende Überlappung und da Paarknoten $(1,2)$ und $(2,3)$ eine gute Schnittgeometrie aufweisen, kann die Konstruktion stattdessen über diese erfolgen. Eine eventuelle Instabilität von $(1,3)$ hat hierbei keine Auswirkungen auf die Tripletkonstruktion (Hartley und Zisserman 2004). Würde $(1,3)$ jedoch keine ausreichende Überlappung aufweisen, könnte man von fehlender Transitivität (vgl. Abschnitt 4.4.1) ausgehen und die Kante zwischen $(1,2)$ und $(2,3)$ würde nicht existieren. Somit ermöglicht der Paargraph die Modellierung sowohl von ungültigen als auch von instabilen Paaren zusammen mit der Bestimmung eines geeigneten Masterbildes für eine zuverlässige Tripletkonstruktion.

Ein zusammenhängender Paargraph ist für eine vollständige Bildverknüpfung nicht notwendig (vgl. Abschnitt 5.6). Umgekehrt impliziert ein zusammenhängender Paargraph keine vollständige

Bildverknüpfung, da einzelne Bilder nicht verknüpft sein können. Die Vollständigkeit einer Bildverknüpfung ist daher erst durch die Erweiterung auf den Verknüpfungsgraphen modellierbar (siehe Abschnitt 4.6).

4.4.1 Transitive Relation zwischen Bildern

Ein Triplet $T = \{a, b, c\}$ erfordert als dreielementige Bildmenge die Existenz einer Äquivalenzrelation $\sim$, sodass eine transitive Relation

$$a \sim b \quad \wedge \quad b \sim c \quad \Rightarrow \quad a \sim c \tag{4.3}$$

zwischen seinen Bildern gegeben ist. Dies impliziert die Existenz von dreifachen Korrespondenzen bzw. einem zweifachen Überlappungsbereich. Ob die Transitivität zwischen den Bildern gegeben ist, kann letztendlich nur über die Dreibildgeometrie festgestellt werden. Ist diese nicht bestimmbar, existiert auch keine Äquivalenzrelation und das zugehörige Triplet ist ungültig. Die Modellierung der Transitivität kann die Anzahl aufwändiger Operationen zur Bestimmung der Dreibildgeometrie reduzieren, indem sie die Konstruktion solcher Triplets bereits im Voraus einschränkt.

Abbildung 4.4: Fehlende transitive Relation zwischen drei Bildern a (links), b (mittig) und c (rechts). Überlappungsbereiche sind mittels farbigen Rechtecken gekennzeichnet.

Im Falle des Bildgraphen, bei dem benachbarte Knoten einen Überlappungsbereich basierend auf ihrer Ähnlichkeit implizieren, ist die transitive Relation nicht immer gegeben. Ein solcher Fall ist in Abb. 4.4 veranschaulicht. Bilder a und b sowie b und c weisen aufgrund großer Überlappungen hohe Ähnlichkeiten auf. Diese wiederum könnten zur Annahme führen, dass a und c ebenfalls einen Überlappungsbereich aufweisen und somit eine dreifache Überlappung zwischen den Bildern existiert. Aufgrund fehlender Transitivität ist dies allerdings nicht gegeben, sodass die Konstruktion eines Triplets nicht möglich ist. Selbst die Forderung nach Dreiecken im Bildgraphen ist nicht anwendbar, da nur binäre Relationen zwischen den Bildern beschrieben werden. Somit eignet sich der Bildgraph nur bedingt zur Modellierung der Bildverknüpfung über Triplets.

Im Gegensatz zum Bildgraphen beschreibt der Paargraph Relationen zwischen den Paaren, die ein gemeinsames Bild beinhalten. Dies erlaubt die Modellierung der Transitivität über die Forderung, dass eine Kante zwischen zwei Paarknoten nur dann existiert, wenn deren Bilder gleiche Äquivalenzklassen besitzen und somit äquivalent sind. Auf diese Weise lassen sich Triplets, deren Bilder über keine transitive Relation verfügen, von der Konstruktion ausschließen.

4.4.2 Gewichtung

Ein ungewichteter Paargraph beschreibt mögliche Verknüpfungen zwischen den Bildern. Über die Gewichtung seiner Kanten lässt sich zudem die geschätzte Qualität der Bildverknüpfung beschreiben. Diese wird im Wesentlichen durch die Qualität der Triplets bestimmt, diese wiederum von der Qualität der sie bildenden Paare. Die Gewichtung des Paargraphen verfolgt somit das Ziel, basierend auf Informationen über Paare geeignete Triplets für die Bildverknüpfung auszuwählen, die mit hoher Wahrscheinlichkeit eine gute Qualität aufweisen.

Die Qualität eines Paares oder Triplets ist vor allem von der Kamerakonfiguration und dem Überlappungsbereich der Bilder abhängig (vgl. Abschnitt 2.1.6). Die Qualität eines Triplets (s_1, m, s_2) mit Masterbild m wird von der Qualität der bildenden Paare (m, s_1) und (m, s_2) sowie der Überlappung zwischen s_1, s_2 und m beeinflusst. Das Paar (s_1, s_2), bestehend aus den Slavebildern, kann hierbei sogar eine kritische Konfiguration aufweisen, ohne dass die Geometrie des Triplets davon negativ betroffen ist (Hartley und Zisserman 2004). Betrachtet man beispielsweise die Paarknoten $(2, 3)$ und $(3, 5)$ im Paargraphen aus Abb. 4.3, so korrespondiert die verbindende Kante mit dem Triplet $(2, 3, 5)$, das das Bild 3 als Masterbild besitzt. Folglich müssen Paare korrespondierend mit den Paarknoten eine gute Qualität aufweisen, während bei mit den Kanten korrespondierenden Paaren lediglich eine ausreichende Überlappung genügt.

Ein Gütemaß für die Stabilität eines Paares wurde in (Beder und Steffen 2006) vorgestellt. Es basiert auf der Schnittgeometrie der rekonstruierten 3D-Punkte, hängt lediglich von der relativen Geometrie der beiden Kameras sowie den Merkmalspositionen ab und ergibt sich für einen 3D-Punkt $\boldsymbol{X}$ über die Rundheit $\mathrm{R}(\boldsymbol{X})$ seines Fehlerellipsoids (siehe Abschnitt 3.2.3).

Allerdings kann auch ein Paar mit hoher Stabilität nur einige wenige Korrespondenzen beinhalten, die unzureichend zur Konstruktion eines Triplets sind. Der Grund hierfür kann beispielsweise in einer großen Basis liegen, die zwar eine gute Schnittgeometrie impliziert, jedoch zu einer unzureichenden Anzahl an Korrespondenzen führt. Auf der anderen Seite ist die Anzahl der Korrespondenzen als eigenständiges Gütemaß ebenfalls nicht ausreichend, weil eine hohe Anzahl kritische Kamerakonfigurationen begünstigen kann. Die Kombination mit der Rundheit eines 3D-Punktes führt hingegen zum Qualitätsmaß

$$\mathrm{Q}_P = \sum_{\boldsymbol{X} \in P} \mathrm{R}(\boldsymbol{X}) \tag{4.4}$$

für ein Paar P, das neben der Korrespondenzanzahl auch ihre Qualität berücksichtigt. Damit besteht die Möglichkeit, eine große Überlappung unter Beachtung einer stabilen Kamerageometrie anzustreben.

Neben der Anzahl der Korrespondenzen beeinflusst auch ihre Verteilung die Qualität eines Triplets (Schönberger und Frahm 2016). In (Zach et al. 2008) und (Irschara et al. 2009) wurden hierzu die Merkmale als Kreise mit einem bestimmten Radius definiert, um ihre Verteilung über die Abdeckung des Bildes abzuschätzen zu können. Gherardi et al. (2010) bestimmten die konvexe Hülle um die Korrespondenzen und teilten diese durch die Bildfläche. Schönberger und Frahm (2016) unterteilten das Bild in ein regelmäßiges Gitter und zählten, wie viele Merkmale in einer Gitterzelle enthalten sind. Um eine genauere Verteilung zu erhalten, erfolgte die gitterförmige Unterteilung auf mehreren Bildauflösungen.

Eine explizite Berücksichtigung der Größe des Überlappungsbereichs und insbesondere der Verteilung der Korrespondenzen im Zusammenhang mit der in dieser Arbeit eingesetzten robusten Kameraposeschätzung ist allerdings schwierig. Einer der Gründe ist, dass die Geometrieschätzung für Paare und Triplets im Rahmen der robusten Kameraposeschätzung auf unterschiedlichen Auflösungen stattfindet (vgl. Abschnitt 5.2). Somit ist die Überlappung und die Verteilung in Paaren nicht ohne Weiteres auf Triplets übertragbar. Auch die Abschätzung der Abdeckungsfläche über die konvexe Hülle wie in (Gherardi et al. 2010) ist aufgrund von Ausreißern unzuverlässig, weil diese eine fälschlicherweise deutlich zu große Fläche implizieren könnten. Stattdessen erfolgt eine indirekte Berücksichtigung der Größe des Überlappungsbereichs indem nur Merkmale, die sich im mehrfachen Überlappungsbereich befinden, betrachtet werden.

Sei M_b die Merkmalsmenge eines Bildes b. Die konvexe Hülle $\mathrm{H}(M_b)$ um die Bildpunkte aus M_b beschreibt den Bereich in b, der von den Merkmalen abgedeckt ist. Merkmalskorrespondenzen eine Paares P bilden die Korrespondenzmenge M_P und $\mathrm{H}(M_P \cap M_b)$ den Überlappungsbereich im Bild $b \in P$ bezüglich des Paares P. Folglich definieren die Korrespondenzen der Paare aus einer Paarmenge Ω den mehrfachen Überlappungsbereich

$$\mathrm{O}(b|\Omega) = \bigcap_{P \in \Omega \wedge P \ni b} \mathrm{H}\left(M_P \cap M_b\right) \tag{4.5}$$

im Bild $b \in P$ mit $P \in \Omega$. Die Qualität des Überlappungsbereichs lässt sich schließlich über die Gleichung (4.4) bezüglich der beinhalteten Bildpunkte $\boldsymbol{x} \mathrel{\widehat{=}} \boldsymbol{X}$ abschätzen:

$$\mathrm{Q}(\Omega|\Omega') = \min_{b \in \bigcup\limits_{P \in \Omega'} P} \left(\sum_{\boldsymbol{x} \in \mathrm{O}(b|\Omega)} \mathrm{R}(\boldsymbol{X}) \right) \tag{4.6}$$

Ω' gibt die Teilmenge der Paare an, für die Gleichung (4.4) anzuwenden ist. Damit besteht die Möglichkeit, den zu betrachtenden Überlappungsbereich getrennt von den zur Bestimmung der Stabilität zu berücksichtigenden Paaren anzugeben.

Basierend auf Gleichung (4.6) wird für eine Kante e mit korrespondierendem Triplet $T = (P_u \cup P_v \cup P_e)$ und Paar P_e sowie Paaren P_u und P_v, die mit den inzidenten Knoten korrespondieren, die Gewichtsfunktion

$$\mathrm{w}(e) = \mathrm{Q}\left(\{P_u, P_v, P_e\} \,|\, \{P_u, P_v\}\right) \tag{4.7}$$

definiert. Da ausschließlich P_u und P_v für die Stabilität von T verantwortlich sind, wird nur deren Qualität betrachtet. Um jedoch auch eine ausreichend Überlappung zwischen den Bildern zu berücksichtigen, erfolgt die Einschränkung der 3D-Punkte von P_u und P_v auf den dreifachen Überlappungsbereich.

4.5 Tripletgraph

Eine explizite Modellierung der Bildverknüpfung über Triplets lässt sich mittels eines zweiten Kantengraphen $L(G_P) = L(L(G_I))$ erzielen. Hierbei entsprechen Triplets den Knoten, die miteinander verbunden sind, falls Triplets zwei gemeinsame Bilder beinhalten. Dies wird demnach als *Tripletgraph* bezeichnet und ist in Abb. 4.3(b) beispielhaft dargestellt. Eine ähnliche Beschreibung

der Beziehungen der Bilder über Triplets erfolgte in (Klopschitz et al. 2010; Jiang et al. 2013; Wefelscheid 2013), wobei Wefelscheid (2013) einen gerichteten Graphen benutzt.

Sowohl Paar- als auch Tripletgraph erlauben eine äquivalente Modellierung. Während der Paargraph geeigneter zur Modellierung der Tripletkonstruktion ausgehend von Bildern und Paaren ist, bietet der Tripletgraph eine intuitivere Beschreibung hinsichtlich der hierarchischen Vereinigung. Da ein Kantengraph eine wesentlich höhere Dichte im Vergleich zu seinem Ursprungsgraphen besitzt, ermöglicht der Paargraph zudem eine wesentlich effizientere und somit geeignetere Repräsentation der Bildverknüpfungen als der Tripletgraph.

4.6 Verknüpfungsgraph

Die Erweiterung des Paargraphen um eine explizite Repräsentation von Bildern mittels eines zweiten Knotentyps führt zum *Verknüpfungsgraphen* $G_L = (V_L, E_L)$ mit $G_L \supset G_P$. Die Knotenmenge $V_L = V_I \cup V_P$ beinhaltet zwei Knotentypen, die mit den Knoten des Bild- und des Paargraphen korrespondieren (siehe Abb. 4.5). Knoten $v \in V_I$ werden als *Bildknoten* und Knoten $v \in V_P$ als *Paarknoten* bezeichnet. Ein Bild- und ein Paarknoten sind benachbart, wenn das zugehörige Paar das korrespondierende Bild enthält. Somit entspricht G_L dem totalen Graphen (Harary 1969) des Bildgraphen ohne Kanten zwischen den Bildknoten.

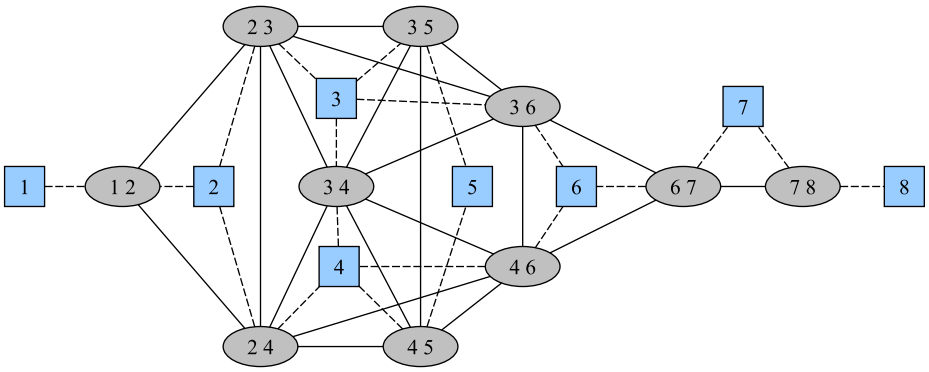


Abbildung 4.5: Verknüpfungsgraph konstruiert aus dem Paargraphen in Abb. 4.3. Bildknoten sind als Rechtecke und Paarknoten als Ellipsen dargestellt.

Der Verknüpfungsgraph dient zur Beschreibung von Verknüpfungen zwischen den Bildern über Triplets. Diese sind zum einen erforderlich, um den Paargraphen zu konstruieren (vgl. Abschnitt 5.4) und zum anderen, um die Blockkonstruktion als ein graphentheoretisches Problem formulieren zu können (siehe Abschnitt 4.7). Abgesehen davon beinhaltet der Verknüpfungsgraph die gleiche Information wie der zugrunde liegende Paargraph.

Die mit den Bildknoten inzidenten Kanten enthalten keine zusätzliche Information, sondern dienen der Auswahl von geeigneten Paarknoten. Daher werden diese mit dem Qualitätsmaß Q_P des Paares P, das mit dem inzidenten Paarknoten korrespondiert, gewichtet (vgl. Abschnitt 4.4.2).

Ein zusammenhängender Verknüpfungsgraph ist ein notwendiges Kriterium für eine vollständige Bildverknüpfung. Jedoch impliziert der Zusammenhang des Verknüpfungsgraphen nicht die Exis-

tenz einer vollständigen Bildverknüpfung. Betrachtet man beispielsweise die Paarknoten $(6,7)$ und $(7,8)$ in Abb. 4.5 mit den zugehörigen Bildknoten als einen eigenständigen, zusammenhängenden Verknüpfungsgraphen und wäre die Transitivität zwischen 6, 7 und 8 nicht gegeben, so würde die Kante zwischen $(6,7)$ und $(7,8)$ nicht existieren. Der Verknüpfungsgraph wäre aber weiterhin zusammenhängend ohne die Möglichkeit einer vollständigen Bildverknüpfung.

4.7 Block

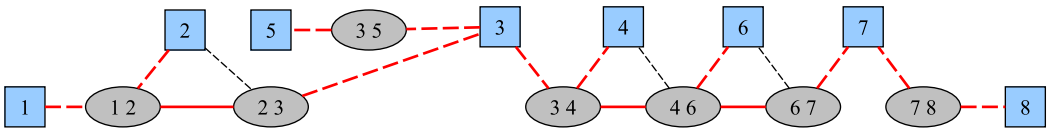
Ein Verknüpfungsgraph beschreibt vollständig die Bildverknüpfungen nach den Anforderungen (A_1) bis (A_5). Darin können unter Umständen auch hochgradig redundante Verknüpfungen vorkommen, die zu keiner signifikanten Verbesserung der Poseschätzung führen, jedoch die Laufzeit signifikant erhöhen. Aus diesem Grund wird das Konzept eines *Blocks* als Verknüpfungsteilgraphen eingeführt, der die essenzielle Anzahl an Verknüpfungen für eine erfolgreiche Kameraposeschätzung enthält. Seine *Größe* beschreibt die Anzahl der verknüpften Bilder und seine *Dichte* die Anzahl der dafür verwendeten Triplets. Ein Block wird als *vollständig* bezeichnet, wenn er alle Bilder einer Bildmenge miteinander verknüpft. Unter *Stabilität* eines Blocks versteht man die Robustheit des Blocks gegenüber kritischen Kamerakonfigurationen, die sich negativ auf die Kameraposeschätzung auswirken können (vgl. Abschnitt 2.1.6).

Die Bestimmung eines Blocks mit minimaler Dichte kann als die Suche nach dem minimalen terminalen Steinerbaum (siehe Abschnitt 2.2.4) mit Bildknoten als Terminalen und Paarknoten als Steinerknoten formuliert werden. Zu beachten ist, dass sich minimale Spannbäume oder Steinerbäume hierfür nicht eignen. Spannbäume ermöglichen nur die Bestimmung einer Kantenteilmenge, während Steinerbäume auch eine Knotenteilmenge in Form von Steinerknoten bestimmen.

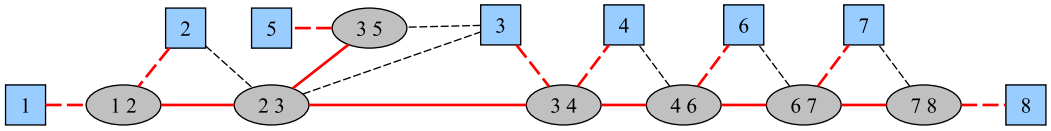
Im Gegensatz zu terminalen Steinerbäumen können bei Steinerbäumen Terminale auch als innere Knoten vorkommen. Im Zusammenhang mit Blöcken kann dies jedoch wie in Abb. 4.6 dargestellt zu Problemen führen. Beim terminalen Steinerbaum erfolgt die Verbindung zwischen Knoten $(2,3)$ und $(3,4)$ über eine Kante, die mit einem gültigen Triplet korrespondiert. Beim Steinerbaum hingegen findet diese Verbindung über den Bildknoten 3 statt. Dies ist unproblematisch falls die Kante zwischen $(2,3)$ und $(3,4)$ im Ähnlichkeitsgraphen existiert. Fehlt diese jedoch, so wäre eine Verbindung über ein ungültiges Triplet hergestellt. Das Gleiche gilt auch für die Verbindungen zwischen Knoten $(2,3)$ und $(3,5)$ sowie $(6,7)$ und $(7,8)$. Somit dürfen Paarknoten ausschließlich direkt verbunden werden, was mittels terminalen Steinerbäumen sichergestellt ist.

4.8 Schleifenschluss

Der Paargraph eines Blocks entspricht graphentheoretisch einem Baum, wodurch eventuell enthaltene Bildschleifen offen sind. Dies kann zu Abweichungen in den geschätzten Kameraposen führen (vgl. Abschnitt 2.1.7), weswegen Bildschleifen explizit detektiert und geschlossen werden müssen.



(a) Steinerbaum



(b) Terminaler Steinerbaum

Abbildung 4.6: Steinerbaum und terminaler Steinerbaum des Verknüpfungsgraphen aus Abb. 4.5. Lediglich rote Kanten gehören zusammen mit den Knoten zu den Steinerbäumen. Schwarze Kanten zwischen Bild- und Paarknoten wurden nachträglich hinzugefügt, um die Form des Verknüpfungsgraphen wiederherzustellen.

4.8.1 Länge der Bildsequenz

Ausgehend von einem Triplet erfolgt die Verknüpfung eines neuen Bildes über ein weiteres Triplet. Somit lässt sich die Länge einer Bildsequenz über die Anzahl der verknüpften Triplets ermitteln. Übertragen auf den Verknüpfungsgraphen ergibt sich die Länge l einer Bildsequenz mit den Endbildern b_1 und b_2 aus der Länge d_L des kürzesten Pfades zwischen den korrespondierenden Bildknoten:

$$\mathrm{l}(b_1, b_2) = \mathrm{d_L}(b_1, b_2) - 2 \tag{4.8}$$

Der Pfad zwischen b_1 und b_2 beinhaltet auch zwei Kanten zwischen den Bild- und Paarknoten, die von seiner Länge abgezogen werden, um die Bildsequenzlänge $\mathrm{l}(b_1, b_2)$ zu erhalten. Diese entspricht dann der Anzahl der Kanten zwischen den Paarknoten, die wiederum mit der Anzahl der Triplets korrespondiert.

Der Block aus Abb. 4.6(b) entspricht beispielsweise einer Bildsequenz mit den Endbildern 1 und 8. Der kürzeste Pfad von 1 nach 8 ist:

$$1 \longrightarrow (1,2) \longrightarrow (2,3) \longrightarrow (3,4) \longrightarrow (4,6) \longrightarrow (6,7) \longrightarrow (7,8) \longrightarrow 8$$

Die Kanten zwischen 1 und $(1,2)$ sowie zwischen $(7,8)$ und 8 tragen nicht zur Bildsequenzlänge bei und werden abgezogen. Dies führt zu einer Bildsequenzlänge von 5 bei 7 Bildern, was als Verknüpfung von 5 neuen Bildern zum Paar $(1,2)$ interpretiert werden kann.

Geometrische Abweichungen innerhalb einer Bildsequenz sind von ihrer Länge abhängig (vgl. Abschnitt 2.1.7). Je länger diese ist, desto größere Abweichungen können auftreten und desto hilfreicher ist ein Schleifenschluss. Bei kurzen Bildsequenzen hingegen führt ein Schleifenschluss zu keinen signifikanten Verbesserungen (Repko und Pollefeys 2005). Vielmehr erhöht er den Rechenaufwand aufgrund der Ermittlung und Verifikation von zusätzlichen Verknüpfungen. Daher wird ein Schwellenwert l_{min} eingeführt, der die *Mindestlänge* einer Bildsequenz festlegt, die als schließbare Bildschleife zu betrachten ist.

4.8.2 Detektion einer Bildschleife

Die Detektion einer Bildschleife lässt sich als Suche nach überlappenden Endbildern einer Bildsequenz formulieren. Die Überlappung stellt hierbei sicher, dass es sich um eine Bildschleife handelt und nicht um eine beliebige Bildsequenz.

Die mit den Endbildern einer Bildsequenz korrespondierenden Bildknoten tendieren dazu, eine hohe Exzentrizität im Verknüpfungsgraphen aufzuweisen. Bei einfachen Bildmengen kann außerdem davon ausgegangen werden, dass die Endbilder den äußeren Knoten des baumartigen Verknüpfungsgraphen entsprechen. Bei komplexeren Bildmengen existieren allerdings auch Konfiguration, bei denen dies nicht notwendigerweise gilt. Abb. 4.7 zeigt eine solche Konfiguration mit den Endbildern 1 und 2. Während 1 über eine hohe Exzentrizität verfügt, besitzt das Endbild 2 aufgrund der zentralen Lage die minimale Exzentrizität.

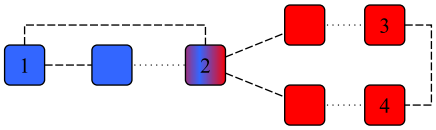


Abbildung 4.7: Bildüberlappungen in zwei Bildsequenzen. Die Überlappungen sind als gestrichelte Kanten dargestellt. Gepunktete Kanten stehen für eine beliebige Menge von Bildern, die zwischen den verbundenen Bildern liegen. Blaue rechteckige Knoten gehören zur ersten Bildsequenz, rote zur zweiten und der rot-blaue zu beiden.

Um Bildschleifen auch in komplexen Konfigurationen zuverlässig ermitteln zu können, ist es erforderlich, jedes Bild als potentielles Endbild zu betrachten. Ausgehend von einem Endbild b_1 beschreibt

$$B_{b_1} = \left\{ b \;\middle|\; \mathrm{l}(b_1,b) \geq \rho l_{min} \wedge \left(\mathrm{d}(b_1,b) \leq d_{b_1} \vee \mathrm{s}(b_1,b) \geq s_{b_1} \right) \wedge \sphericalangle(b_1,b) \leq \alpha \right\} \tag{4.9}$$

die Menge der geeigneten Bilder für das zweite Endbild b_2. $\mathrm{l}(b_1,b)$ gibt die Länge der Bildsequenz zwischen b_1 und b an, wobei l_{min} der Mindestlänge und $\rho \geq 1$ einem beliebigen Skalierungsfaktor entspricht. d steht für die euklidische Distanz und $\sphericalangle$ für den Blickwinkelunterschied zwischen den Kameraposen der Bilder. Über s erfolgt die Abschätzung der Ähnlichkeit zwischen b_1 und b.

Schließlich wird als das geeignetste Endbild b_2 ein Bild $b \in B^*_{b_1} \subseteq B_{b_1}$ betrachtet, das mit b_1 die längste Bildsequenz bildet und zudem möglichst nah zu b_1 liegt:

$$B^*_{b_1} = \underset{b \in B_{b_1}}{\arg\max}\, \mathrm{l}(b_1,b) \quad \Rightarrow \quad b_2 = \underset{b \in B^*_{b_1}}{\arg\min}\, \mathrm{d}(b_1,b)\,. \tag{4.10}$$

Da in B_{b_1} nur räumlich nahe bzw. ähnliche Bilder zu b_1 enthalten sind, handelt es sich beim gewählten b_2 mit hoher Wahrscheinlichkeit um ein Endbild, das mit b_1 eine Bildschleife bildet. Ob b_1 und b_2 tatsächlich zwei Endbilder einer Bildschleife sind, hängt letztendlich vom Erfolg ihrer Verknüpfung ab (siehe Abschnitt 4.8.3).

Räumliche Suche

Eine höhere Wahrscheinlichkeit einer Überlappung und somit der Ausschluss von ungeeigneten Endbildern kann durch Einbeziehung der (grob) geschätzten Kameraposen erzielt werden. Letztere lassen sich effizient mittels hierarchischer Vereinigung ohne Anwendung der Bündelausgleichung ermitteln (siehe Abschnitt 5.7). Basierend darauf gibt $\mathrm{d}(b_1, b)$ in Gleichung (4.9) den euklidischen Abstand zwischen den Kamerapositionen und $\sphericalangle$ den Blickwinkelunterschied an. Die Einschränkung des Blickwinkelunterschiedes mittels α dient vor allen dazu, in die entgegengesetzte Blickrichtung aufgenommene Bilder auszuschließen.

Der Schwellenwert d_{b_1} sichert die räumliche Nähe und erhöht damit die Wahrscheinlichkeit einer Überlappung. Da d_{b_1} von der Umgebung von b_1 abhängt, ist die Verwendung eines einheitlichen Schwellenwerts nicht möglich. Dieser wäre für unterschiedliche Bildmengen schon aufgrund des unbekannten Maßstabs ungeeignet. Auch innerhalb einer Bildmenge ist er nicht anwendbar, weil die Bildabstände in den Umgebungen von Bildern stark variieren können. Seien beispielsweise zwei Bilder b_t und b_u gegeben, wobei b_t aus einer Bildsequenz um eine Hausecke stammt und b_u aus einem Überflug einer Drohne. Um eine ausreichende Überlappung sicherzustellen, müssen die Abstände der Bilder um die Hausecke gering sein, während bei einem Überflug deutlich größere Abstände möglich sind. Folglich befinden sich in einer bestimmten Umgebung von b_t deutlich mehr Bilder als dies in der gleich großen Umgebung von b_t der Fall ist. Ein einheitlicher Schwellenwert würde hier entweder zu viele oder nicht genug Bilder einbeziehen.

Eine geeignete Umgebung für b_1 lässt sich aus den kürzesten Pfaden von b_1 zu anderen Bildknoten im Verknüpfungsgraphen ableiten. Die Pfadlänge l kann als Indiz für den euklidischen Abstand hergenommen werden, wobei eine größere Pfadlänge nicht einen größeren euklidischen Abstand impliziert. Begrenzt man die betrachtete Umgebung um b_1 mittels einer maximalen Pfadlänge l_d, so lässt sich d_{b_1} aus den euklidischen Abständen zu den Bildern, die auf den Pfaden liegen, ermitteln:

$$d_{b_1} = \max_{\forall b : \mathrm{l}(b_1, b) \leq l_d} \mathrm{d}(b_1, b) \tag{4.11}$$

Um einen sinnvollen Wert für d_{b_1} zu erhalten, ist es notwendig, lediglich Pfade zu den Bildern aus der näheren Umgebung von b_1 zu betrachten. Diese kann über einen einheitlichen Schwellenwert l_d festgelegt werden, der im Vergleich zur d_{b_1} unabhängig von der Aufnahmekonfiguration ist. Für $l_d = 1$ ergibt sich d_{b_1} aus den Abständen zu den Bildern, die sich im gleichen Triplet wie b_1 befinden. Wenn die Umgebung von b_1 dicht ist, kann $l_d = 1$ dazu führen, dass d_{b_1} eine zu kleine Umgebung definiert. Verwendet man stattdessen $l_d = 5$, so werden auch Bilder betrachtet, die über 5 weitere Bilder mit b_1 verknüpft wurden. Die Wahrscheinlichkeit, dass ausschließlich Bilder aus der näheren Umgebung von b_1 stammen, verringert sich damit.

Ähnlichkeitssuche

Bei langen Bildsequenzen kann es zu starken geometrischen Abweichungen kommen, die fälschlicherweise zu sehr großen Abständen zwischen den geschätzten Kameraposen der Endbilder führen können. Die räumliche Suche ist in diesem Fall aufgrund des zu niedrigen Schwellenwerts d_{b_1} nicht in der Lage betroffene Endbilder zu ermitteln. Die Erhöhung von d_{b_1} würde hingegen zu unnötigem Anstieg an Komplexität führen.

Stattdessen erfolgt die Ermittlung potenzieller Endbilder über die Bildähnlichkeiten, indem für ein Endbild b_1 zusätzlich alle Bilder mit einer Ähnlichkeit von mindestens s_{b_1} als zweites Endbild b_2 berücksichtigt werden. Der Schwellenwert s_{b_1} in Gleichung (4.9) ergibt sich aus der minimalen Ähnlichkeit zwischen Bildern, die im Verknüpfungsgraphen des Blocks mit b_1 im gleichen Paarknoten enthalten sind.

Während bei der räumlichen Suche die Mindestlänge einer Bildschleife unverändert bleibt (d. h. $\rho = 1$), wird diese über den Skalierungsfaktor $\rho > 1$ in Gleichung (4.9) für die Ähnlichkeitssuche erhöht, um die Anzahl an potenziellen Endbildern zu begrenzen. Da starke Abweichungen in Kameraposen nur bei sehr langen Bildsequenzen auftreten können und die Ähnlichkeitssuche ausschließlich für solche Fälle konzipiert ist, stellt die Erhöhung der Mindestlänge eine sinnvolle Vorgehensweise zur Reduktion der Komplexität dar. Auf diese Weise werden nur ähnliche Bilder, die jedoch im Verknüpfungsgraphen weit voneinander entfernt sind, als mögliche Endbilder einer Bildschleife betrachtet.

4.8.3 Schließen einer Bildschleife

Das Schließen einer Bildschleife erfolgt über die Verknüpfung ihrer Endbilder. Da die eingesetzte hierarchische Vereinigung nach (Mayer 2014) auf den eindeutigen Identifikationsnummern der Merkmalspunkte basiert, genügt es für den Schleifenschluss lediglich die Endbilder zu verknüpfen, ohne explizit einen Kreis im Paargraphen bilden zu müssen.

Für die Verknüpfung der Endbilder b_1 und b_2 wird ein Triplet $T = (b_1, b_2, b)$ benötigt. Ein Schleifenschluss ist umso hilfreicher, je mehr Verknüpfungen zwischen den Merkmalspunkten der Endbilder hergestellt werden können. Aus diesem Grund und zur Reduzierung der Komplexität wird davon ausgegangen, dass das Paar (b_1, b_2) ähnliche Eigenschaften wie andere Paare, die eines der Endbilder enthalten, aufweist. Somit muss nicht nach schwierigeren Verknüpfungen, als bereits in der Umgebung der Endbilder vorhanden sind, gesucht werden. Übertragen auf den Verknüpfungsgraphen bedeutet dies, dass für (b_1, b) bzw. (b_2, b) ein Paarknoten des Verknüpfungsgraphen benutzt werden kann und somit ein b zu finden ist, das die Konstruktion von T ermöglicht.

Betrachtet man beispielsweise die Bildschleife in Abb. 4.8, die über die Endbilder 1 und n geschlossen werden kann. Um n mit 1 zu verknüpfen, werden die Nachbarknoten von 1, die den Parknoten $(1,2)$, $(1,3)$ und $(1,4)$ entsprechen, betrachtet. Diejenigen Paarknoten, die mit n ein gültiges Triplet bilden, können zum Schleifenschluss eingesetzt werden.

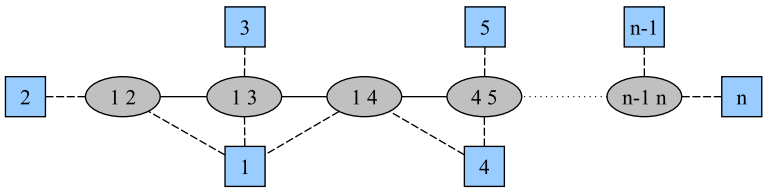


Abbildung 4.8: Verknüpfungsgraph für eine Bildschleife mit den Endbildern 1 und n. Die gepunktete Kante zwischen den Paarknoten steht für eine beliebige Anzahl an Paarknoten dazwischen.

4.8.4 Detektion relevanter Bildschleifen

Ein Block kann mehrere Bildschleifen enthalten, wobei einige davon Teilschleifen einer größeren sein können. In Abb. 4.9 ist eine Aufnahmekonfiguration dargestellt, die mehrere Bildschleifen enthält. Als längste Bildschleife könnte $1-2-...-10-11-...-19-20$ mit den Endbildern 1 und 20 betrachtet werden. Diese enthält wiederum Teilschleifen, wie beispielsweise $1-2-...-5-16-17-...-19-20$. Tatsächlich existieren viele Teilschleifen aufgrund von Überlappungen zwischen benachbarten Bildern. In solchen Fällen wäre das Schließen aller möglichen Teilschleifen sehr aufwändig, da viele zusätzliche Verknüpfungen gefunden und hergestellt werden müssten. Allerdings könnte auch das alleinige Schließen der größten Bildschleife unzureichend sein, um geometrische Abweichungen in längeren Bildteilschleifen zu korrigieren. Somit wird angestrebt, lediglich diejenigen Teilschleifen zu schließen, die für die Korrektur der geometrischen Abweichungen essenziell sind.

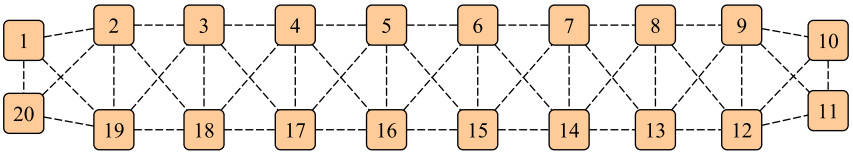


Abbildung 4.9: Überlappungen zwischen Bildern einer Bildschleife dargestellt durch gestrichelte Verbindungslinien. Hierbei könnte es sich beispielsweise um Luftaufnahmen eines Gebietes handeln, wobei während der Bildaufnahme in zwei entgegengesetzte Richtungen geflogen wurde. Somit existieren Überlappungen zwischen allen direkt benachbarten Bildern.

Auf Grundlage der Mindestlänge l_{min} einer Bildschleife (vgl. Abschnitt 4.8.1) lassen sich die längsten unabhängigen Bildschleifen detektieren. Basierend darauf werden auch Teilschleifen, die über die Mindestlänge verfügen, in Betracht gezogen. Zur Reduktion der Anzahl der Teilschleifen wird die Gewichtsfunktion

$$\mathrm{w}(c_e) = e^{-fc_e^2} \tag{4.12}$$

für die Kanten des Paargraphen eingeführt. Diese verringert das Kantengewicht in Abhängigkeit von der Anzahl c_e an Bildschleifen, in denen die Kante e vorkommt. Auf diese Weise verlieren Teilschleifen geschlossener Bildschleifen immer mehr an Relevanz.

Die Verwendung der Exponentialfunktion ermöglicht eine kontinuierliche Dämpfung in einem wohldefiniertem Wertebereich. Der Verlauf der Kantengewichtsfunktion w für verschiedene Werte des Dämpfungsfaktors f ist in Abb. 4.10 dargestellt. Ein Wert von $f = 0$ deaktiviert die Gewichtsfunktion, wodurch alle Teilschleifen mit einer Mindestlänge l_{min} geschlossen werden. Ab etwa $f = 10$ wird das Kantengewicht nach dem Schließen der ersten Bildschleife bereits so stark verringert, dass keine Teilschleife mehr berücksichtigt wird. Mit Dämpfungsfaktoren zwischen 0 und 1 lässt sich daher die Anzahl der zu schließenden Teilschleifen steuern, d. h. je kleiner f, desto mehr Teilschleifen werden geschlossen.

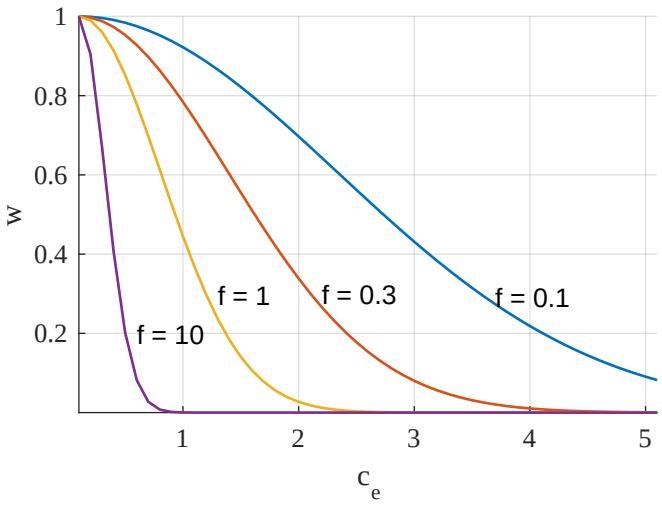


Abbildung 4.10: Kantengewichtsfunktion w in Abhängigkeit von der Anzahl c_e an geschlossenen Bildschleifen für verschiedene Dämpfungsfaktoren f.

5 Automatische Kameraposeschätzung

Basierend auf den theoretischen Grundlagen aus Kapitel 4 befasst sich dieses Kapitel mit der praktischen Umsetzung der automatischen Kameraposeschätzung. Die Zielsetzung wurde im Abschnitt 1.1 sowie Kapitel 4 beschrieben und besteht zusammenfassend darin, eine möglichst vollständige Verknüpfung zwischen den Bildern einer Bildmenge herzustellen, die zudem robust hinsichtlich komplexer Aufnahmekonfigurationen ist.

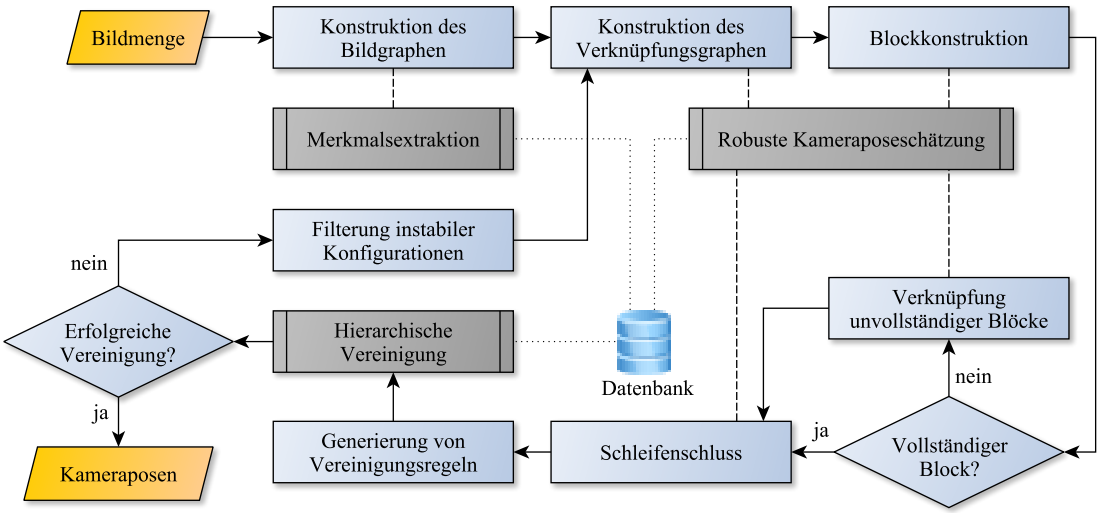


Abbildung 5.1: Verarbeitungsschritte der automatischen Kameraposeschätzung

Die Verarbeitungsschritte der automatischen Kameraposeschätzung sind in Abb. 5.1 dargestellt. Die Grundlage für geometrische Berechnungen bilden die robuste Kameraposeschätzung (Mayer et al. 2012) und die hierarchische Vereinigung (Mayer 2014), gekennzeichnet durch graue Rechtecke. Diese interagieren mit einer Datenbank, die zur Speicherung von Zwischenergebnisse wie beispielsweise Bildpyramiden dient. Die durch blaue Rechtecke dargestellte Verarbeitungsschritte sind Bestandteile dieser Arbeit und das Thema dieses Kapitels.

Ausgehend von einer gegebenen Bildmenge werden als erstes Merkmale in den Bildern extrahiert. Basierend darauf erfolgt die Konstruktion des Bildgraphen (Abschnitt 5.3) und anhand dieses die Konstruktion des Verknüpfungsgraphen (Abschnitt 5.4). Anschließend wird im Rahmen der Blockkonstruktion eine minimale Teilmenge der Triplets für eine effiziente Bildverknüpfung ermittelt (Abschnitt 5.5). Der Verknüpfungsgraph und die Blöcke enthalten ausschließlich verifizierte Paare und Triplets (Abschnitt 5.2) mit stabilen Kamerakonfigurationen (Abschnitt 5.1). Im Falle von unvollständigen Blöcken wird versucht, diese durch gezielte Suche nach Verknüpfungen zu verbinden (Abschnitt 5.6). Darauf folgend werden über Schleifenschlüsse zusätzliche Bildverknüpfungen hergestellt, um die Blockstabilität zu verbessern (vgl. Abschnitt 4.8). Durch die

Vereinigung der Bildteilmengen erfolgt schließlich die Schätzung der relativen Kameraposen (Abschnitt 5.7). Konnte die Vereinigung nicht erfolgreich abgeschlossen werden, so existieren instabile Verknüpfungen, die herauszufiltern sind (Abschnitt 5.8), um eine erfolgversprechende erneute Blockkonstruktion durchführen zu können.

5.1 Detektion kritischer Kamerakonfigurationen

In (Michelini und Mayer 2014) wurde gezeigt, dass die Detektion von Paaren mit Bewegungsdegeneration (vgl. Abschnitt 2.1.6) auf Grundlage der Eigenschaften der 3D-Punkte erfolgen kann. Dazu erfolgte eine binäre Klassifikation in Paare mit und ohne Bewegungsdegeneration. Für eine robuste Kameraposeschätzung ist es aber wichtig, auch Paare mit unzureichender Basis zu ermitteln. Anhand der Eigenschaft der 3D-Punkte und empirisch festgelegter Schwellenwerten wurden in (Michelini und Mayer 2016) die Paare in stabile, instabile und kritische unterteilt. Durch Kombination der Ansätze von (Michelini und Mayer 2014) sowie (Michelini und Mayer 2016) und die Verwendung eines robusten Klassifikators zur Klassifizierung der Paare ist eine Verbesserung in der Detektion kritischer Kamerakonfigurationen zu erwarten.

Die robuste Kameraposeschätzung liefert für jedes Paar die Unsicherheit der geschätzten Kamerapose. Zudem wird für jeden rekonstruierten 3D-Punkt $\boldsymbol{p}$ auch seine Unsicherheit in Form der Kovarianzmatrix $\boldsymbol{C}$ bestimmt. Das zugehörige Fehlerellipsoid ist definiert durch

$$(\boldsymbol{x}-\boldsymbol{p})^T\boldsymbol{C}^{-1}(\boldsymbol{x}-\boldsymbol{p}) = 1, \tag{5.1}$$

wobei $\boldsymbol{x}$ ein Punkt auf dem Ellipsoid ist. Die Eigenvektoren von $\boldsymbol{C}$ definieren die Richtungen der Halbachsen des Ellipsoids und die Eigenwerte $\lambda_1 \geq \lambda_2 \geq \lambda_3$ die Quadrate ihrer Längen a, b und c, d. h.

$$a = \sqrt{\lambda_1} \quad b = \sqrt{\lambda_2} \quad c = \sqrt{\lambda_3}\,. \tag{5.2}$$

Basierend auf der Form des Fehlerellipsoids lassen sich folgende Eigenschaften für einen rekonstruierten 3D-Punkt definieren:

- *Rundheit* des Fehlerellipsoids nach (Beder und Steffen 2006), die zur Gewichtung des Paargraphen benutzt wurde (vgl. Abschnitt 4.4.2):

$$R_p = \sqrt{\frac{\lambda_3}{\lambda_1}} \tag{5.3}$$

- *Volumen* des Fehlerellipsoids nach (Michelini und Mayer 2014):

$$V_p = \frac{4}{3}\pi abc = \frac{4}{3}\pi\sqrt{\lambda_1\lambda_2\lambda_3} = \frac{4}{3}\pi\sqrt{\det(\boldsymbol{C})} \tag{5.4}$$

- *Alternative Rundheit* des Fehlerellipsoids nach (Michelini und Mayer 2014) als Quotient von V_p und Volumen V_p^K der umschreibenden Kugel:

$$K_p = \frac{V_p}{V_p^K} = \frac{\frac{4}{3}\pi abc}{\frac{4}{3}\pi r^3} = \frac{abc}{r^3} = \frac{abc}{max(a,b,c)^3} = \frac{\sqrt{\lambda_2\lambda_3}}{\lambda_1} \tag{5.5}$$

- *Tiefe* D_p des Punktes

Eigenschaften V_p und K_p haben gegenüber R_p den Vorteil, dass sie die komplette Form des Fehlerellipsoids berücksichtigen. Die Motivation für die Verwendung der Punkttiefe D_p liegt in der Tatsache, dass diese proportional zur Basis ist und sich signifikant in Abhängigkeit von der Kamerakonfiguration ändern sollte. Kameras in gültigen Paaren mit geringen Basen müssen eine geringe Rotation aufweisen, um genügend Überlappung sicherzustellen. Dadurch ergeben sich kleine Schnittwinkel, die zusammen mit geringen Basen zu größeren Tiefen der rekonstruierten 3D-Punkte (relativ zur Basis) führen.

Die Klassifizierung der Paare erfolgt mittels Random Forest (Abschnitt 2.5.3) basierend auf folgenden Klassifikationsmerkmalen, die aus Eigenschaften über alle 3D-Punkte gebildet wurden:

- Median der Rundheiten R
- Median der Volumen V
- Median der alternativen Rundheiten K
- Median der Punkttiefe D

Als weitere Merkmale werden noch der mittlere Rückprojektionsfehler *MRE*, die Inlier-Rate *IR* und die höchste Unsicherheit T in den Koordinaten der geschätzten Translationsrichtung für ein Paar verwendet. Die Unsicherheit der geschätzten Rotation ist hingegen kein geeignetes Merkmal, weil diese selbst bei geringen Basen zuverlässig bestimmt werden kann (Enqvist et al. 2011).

Im Abschnitt 6.3 wird die Vorgehensweise zur Klassifizierung beschrieben und gezeigt, dass diese im Vergleich zur Verwendung einzelner Schwellenwerte (Michelini und Mayer 2016), eine deutlich zuverlässigere Detektion von kritischen Kamerakonfigurationen bei den Paaren ermöglicht.

Im Falle von Triplets erfolgt die Unterteilung gemäß Abschnitt 4.2 anhand der sie bildenden Paare (vgl. Abschnitt 2.1.5). Sind diese kritisch oder instabil, wird auch das Triplet als solches betrachtet. Zudem wird das Verhältnis der Längen der Basen der ein Triplet bildenden Paare berücksichtigt. Ist dieses zu groß, wird das Triplet als instabil betrachtet.

5.2 Verifikation von Paaren und Triplets

Die Überprüfung der Kamerageometrie von Paaren und Triplets erfolgt im Rahmen der *Verifikation* anhand des Algorithmus 5.1. Dazu wird die Kamerapose erstmal mittels robuster Kameraposeschätzung nach (Mayer et al. 2012) auf einer festgelegten Bildauflösung s geschätzt (Zeile 2). Paare und Triplets mit einer unzureichenden Anzahl an Korrespondenzen werden als ungültig betrachtet. Die restlichen werden auf kritische Kamerakonfiguration überprüft (Zeile 3). Das Ergebnis der Verifikation sind die Teilmengen von Paaren bzw. Triplets entsprechend der Unterteilung im Abschnitt 4.2 (Zeile 10).

Die Angabe der Bildauflösung s dient zur Reduktion des Rechenaufwandes der robusten Kameraposeschätzung. Bei Paaren führt eine niedrige Auflösung zu ein paar hundert Merkmalen pro Bild. Diese sind gewöhnlich für eine erfolgreiche robuste Kameraposeschätzung von Paaren mit

Eingabe: Elementmenge $\mathcal{E}$, Bildauflösung s, Verknüpfungsgraph $G_L = (V_L, E_L)$, Bildgraph $G_I = (V_I, E_I)$
Ausgabe: Elementteilmengen nach Abschnitt 4.2

```
1  function Verifikation(E, s, G_L, G_I)
2      E+, E- ← RobusteKameraposeschätzung(E, s)
3      E*, E×, E~ ← Klassifizierung(E+)                    siehe Abschnitt 5.1
4      if V_L ≠ ∅ then
5          G_L ← G_L ⊖ (E \ E*)                            ⊖ Graphreduktion
6      end
7      if V_I ≠ ∅ then
8          G_I ← G_I ⊖ (E \ E*)
9      end
10     return E*, E×, E~, E-
11 end
```

Algorithmus 5.1: Verifikation für Paare und Triplets mit Graphreduktion $\ominus$

moderaten Basen ausreichend. Da solche Paare weitaus häufiger vorkommen als beispielsweise solche mit großen Basen, wird durch die Verwendung einer niedrigen Auflösung die Effizienz deutlich gesteigert. Allerdings kann diese für Paare mit großen Verzerrungen unzureichend sein, weswegen in diesen Fällen die Verwendung einer höheren Bildauflösung nötig ist.

Im Falle von Triplets ist die Bildauflösung für eine erfolgreiche Kameraposeschätzung weniger kritisch. Da die Kameraposeschätzung auf der Grundlage der Geometrie der sie bildenden Paaren stattfindet (vgl. Abschnitt 2.1.5), kann der Suchraum deutlich eingeschränkt und somit bereits zu Beginn eine höhere Bildauflösung als bei den Paaren verwendet werden. Somit dient hier die Erhöhung der Auflösung lediglich der Ermittlung von zusätzlichen Korrespondenzen bei bereits gültigen Triplets.

Der letzte Schritt der Verifikation besteht in der Aktualisierung des Verknüpfungs- und des Bildgraphen (Zeilen 5 und 8). Dies erfolgt über die sogenannte *Graphreduktion* $\ominus$, bei der Knoten und Kanten korrespondierend mit den Elementen aus der angegebenen Elementmenge aus dem Graphen entfernt werden. Im Algorithmus 5.1 sind dies nicht stabile Elemente. Im Falle von Paaren werden korrespondierende Kanten aus G_I sowie Knoten und Kante aus G_L entfernt. Entspricht $\mathcal{E}$ der Tripletmenge, werden lediglich korrespondierende Kanten aus G_L entfernt.

5.3 Konstruktion des Bildgraphen

Die Konstruktion des Bildgraphen basiert auf den paarweisen Bildähnlichkeiten (vgl. Abschnitt 4.1.2). Eine Kante e zwischen zwei Knoten im Bildgraphen existiert nur, wenn $\mathrm{J}(e) > 0$ gilt, mit J dem Jaccard-Index aus Gleichung (4.2) auf S. 53. Dieser Schwellenwert dient lediglich einer groben Reduktion der Graphendichte und somit der Komplexität der nachfolgenden Verarbeitungsschritte. Bilder mit $\mathrm{J} = 0$ werden als unähnlich betrachtet und als potentielle Paare

ausgeschlossen. Ein höherer Schwellenwert könnte hingegen, vor allem bei größeren Bildverzerrungen, zu nicht zusammenhängendem Bildgraphen und folglich zu unvollständiger Bildverknüpfung führen.

5.4 Konstruktion des Verknüpfungsgraphen

Der Verknüpfungsgraph $G_L = (V_L, E_L)$ ist durch den Paargraphen definiert und erweitert diesen um Bildknoten. Der Paargraph $G_P = (V_P, E_P)$ entspricht wiederum dem Kantengraphen des Bildgraphen $G_I = (V_I, E_I)$ und lässt sich anhand Gleichung (2.17) auf S. 33 aus diesem ableiten. Aus dem Zusammenhang in Gleichung (2.18) auf S. 33 sowie $|V_P| = |E_I|$ ist ersichtlich, dass diese Vorgehensweise zu einem sehr großen und dichten Paargraphen führt. Dieser würde zum einen eine speicherintensive Repräsentation erfordern und zum anderen die Komplexität der Graphenalgorithmen erheblich erhöhen. Des Weiteren sind nur Paarknoten in G_P sinnvoll, die mittels robuster Kameraposeschätzung geometrisch verifiziert wurden.

Daher erfolgt der Aufbau des Paargraphen iterativ über maximale Spannbäume des Bildgraphen. Die Motivation dahinter ist, dass bei geeigneten Kamerakonfigurationen bereits ein maximaler Spannbaum zur Herstellung von vollständigen Bildverknüpfungen mit guter Überlappung ausreichend sein kann. Obwohl dies bei komplexen Bildmengen eher selten gegeben ist, eignet sich ein maximaler Spannbaum, um Paare mit potentiell größter Überlappung auszuwählen.

Die Konstruktion ist im Algorithmus 5.2 zusammengefasst. Als Eingabe dient der Bildgraph G_I, der zusammen mit dem Verknüpfungsgraphen G_L einer stetigen Veränderung unterliegt. Zu Beginn enthält G_L lediglich die Bildknoten (Zeile 2). Davon ausgehend werden iterativ Paarknoten hinzugefügt, solange die notwendigen Voraussetzungen zur Konstruktion eines vollständigen Blocks nicht gegeben oder bis weitere Verknüpfungen unwahrscheinlich sind (Zeilen 7 bis 18).

Das Symbol $\oplus$ bezeichnet die sogenannte *Grapherweiterung*, d. h. die Erweiterung eines Graphen um neue Knoten und Kanten. Dabei werden sowohl Paarknoten zu G_L hinzugefügt, die mit den Paaren in der angegebenen Paarmenge korrespondieren, als auch Verbindungen mit den bereits enthaltenen Knoten hergestellt. Die Operation $\mathcal{P}(G)$ liefert hierbei eine Menge $\mathcal{P}_G$ von Paaren korrespondierend mit den Knoten und Kanten des Graphen G.

Jede Iteration startet mit der Konstruktion eines maximalen Spannbaumes in G_I (Zeile 9). Seine Kanten korrespondieren mit den Paarknoten, die zum G_L hinzugefügt werden (Zeile 10). Daraus ergibt sich die Menge $\mathcal{P}(G_L)$ der Paare, die mit den hinzugefügten Knoten sowie sich daraus ergebenden Kanten in G_L korrespondieren. Der maximale Spannbaum ermöglicht die Auswahl der Paare mit der größten Überlappung, sodass dreifache Überlappungen zwischen Bildern korrespondierend mit inzidenten Knoten von benachbarten Kanten wahrscheinlich sind. Das Ziel besteht darin, Paare auszuwählen, die zu Triplets zusammengefasst werden können. Allerdings erhöht diese Vorgehensweise im Falle von kritischen Kamerakonfigurationen die Gefahr der Einbeziehung von instabilen Paaren. Unter Berücksichtigung der vorhandenen Informationen (d. h. ohne Zusatzwissen) stellt dies jedoch die geeignetste Vorgehensweise dar, um grundlegende Bildverknüpfungen herzustellen.

Eingabe: Bildgraph $G_I = (V_I, E_I)$
Ausgabe: Verknüpfungsgraph $G_L = (V_L, E_L)$

1 **if** $|V_L| = \emptyset$ **then**
2 $\quad V_L \leftarrow V_I$ *Initialisierung mit Bildknoten*
3 **end**
4 $\mathcal{P}_V = \emptyset$
5 $n_L \leftarrow |\texttt{Komponenten}(G_L)|$ *siehe Abschnitt 2.2.2*
6 $n_P \leftarrow |\texttt{Komponenten}(G_P)|$ *Paargraph $G_P \subset G_L$*
7 **do**
8 $\quad$ **do**
9 $\quad\quad E \leftarrow \texttt{MaximalerSpannbaum}(G_I)$ *siehe Abschnitt 2.2.4*
10 $\quad\quad G_L \leftarrow G_L \oplus \mathcal{P}(E)$ *$\oplus$ Grapherweiterung*
11 $\quad\quad \mathcal{P}^{\times}, \mathcal{P}^{\sim}, \mathcal{P}^{-} \leftarrow \texttt{Verifikation}(\mathcal{P}(G_L),\ s_1,\ G_L,\ G_I)$ *siehe Abschnitt 5.2*
12 $\quad\quad \mathcal{P}_V \leftarrow \mathcal{P}_V \cup \mathcal{P}^{-} \cup \mathcal{P}^{\sim}$
13 $\quad\quad n \leftarrow n_P$
14 $\quad\quad n_P \leftarrow |\texttt{Komponenten}(G_P)|$
15 $\quad$ **while** $\mathcal{P}^{\times} \neq \emptyset$ **or** $(\mathcal{P}^{-} \neq \emptyset$ **and** $n_P < n)$
16 $\quad n \leftarrow n_L$
17 $\quad n_L \leftarrow |\texttt{Komponenten}(G_L)|$
18 **while** $1 < n_L < n$
19 $\texttt{Verlinkung}(G_I,\ G_L,\ \mathcal{P}_V,\ s_2)$ *$s_2 > s_1$, siehe Algorithmus 5.3*
20 $\texttt{Verdichtung}(G_I,\ G_L,\ s_1)$ *siehe Algorithmus 5.4*

Algorithmus 5.2: Konstruktion des Verknüpfungsgraphen

Die Geometrie der Paare aus $\mathcal{P}(G_L)$ wird verifiziert, um diejenigen mit ungeeigneten Kamerakonfigurationen herauszufiltern (Zeile 11). Die Entfernung dieser aus G_I anhand Algorithmus 5.1 verhindert ihre erneute Auswahl durch den maximalen Spannbaum in der nächsten Iteration. Schließlich verbleiben nur Knoten und Kanten in G_L, die mit den stabilen Paaren korrespondieren.

Anschließend erfolgt die Bestimmung der Anzahl der Zusammenhangskomponenten von G_L und seinem Paargraphen G_P (Zeilen 14 und 17). Notwendige Bedingungen für einen vollständigen Block sind zusammenhängender G_L sowie sein Paargraph G_P (vgl. Abschnitte 4.4 und 4.6). Wenn eine dieser Bedingungen nicht erfüllt ist, wird erneut ein maximaler Spannbaum bestimmt, um weitere Bildverknüpfungen zu ermitteln.

Die innere Schleife (Zeilen 8 bis 15) dient zur Überprüfung von G_P. Eine weitere Iteration ist nur sinnvoll, wenn ungültige, kritische oder instabile Paare auftreten, weil nur in diesem Fall andere Paare durch den maximalen Spannbaum ausgewählt werden. Bei instabilen Paaren ist davon auszugehen, dass in der nächsten Iteration alternative Paare gefunden werden können. Bei ungültigen Paaren können diese hingegen aufgrund von großen Bildverzerrungen und geschätzten Bildähnlichkeiten schwer ermittelbar sein (vgl. Abschnitt 5.6). Um die Suche nach diesen schwierigen Verknüpfungen zu begrenzen, erfolgt eine neue Iteration nur dann, wenn sich die

Anzahl der Zusammenhangskomponenten von G_P reduziert.

Die äußere Schleife (Zeilen 7 bis 18) überprüft den Verknüpfungsgraphen G_L und bricht die Iterationen ab, wenn G_L zusammenhängend ist oder die Anzahl der Zusammenhangskomponenten von G_L unverändert bleibt. Analog zur inneren Schleife wird hierdurch eine lange Suche nach schwierigen Verknüpfungen vermieden.

5.4.1 Verlinkung

Der bis hierhin konstruierte Verknüpfungsgraph besteht ausschließlich aus stabilen Paaren. Dadurch wird zwar die Robustheit gegenüber kritischen Kamerakonfigurationen erhöht, es kann aber auch zu unvollständiger Bildverknüpfung führen, wenn G_L oder G_P nicht zusammenhängend ist. In diesem Fall lassen sich fehlende Verknüpfungen unter Umständen über eine erneute Verifikation von kritischen und ungültigen Paaren ermitteln (Zeile 19 im Algorithmus 5.2).

Die Verifikation der Paare erfolgte bis zu diesem Punkt aus Effizienzgründen auf der reduzierten Bildauflösung s_1 (vgl. Abschnitt 5.2). Mittels einer erneuter Verifikation von kritischen und ungültigen Paaren auf einer höheren Auflösung s_2 lassen sich unter Umständen weitere stabile Paare ermitteln. Im Falle von kritischen Paaren führt die Erhöhung der Auflösung zur genaueren Klassifikation (vgl. Abschnitt 6.3), wodurch sich die Möglichkeit ergibt grenzwertige kritische Paare als stabile zu klassifizieren.

Bilder von ungültigen Paaren überlappen sich entweder tatsächlich nicht oder die Überlappung konnte aufgrund einer zu geringen Bildauflösung s_1 nicht ermittelt werden. Somit kann im letzteren Fall die Verwendung einer höheren Bildauflösung s_2 zur Ermittlung der notwendigen Korrespondenzen führen. Die Erhöhung der Auflösung und erneute Verifikation für alle ungültigen Paare wäre allerdings sehr aufwändig, weil einige tatsächlich keine Überlappung aufweisen. Daher werden nur diejenigen Paare erneut betrachtet, die unter s_1 eine Mindestanzahl an Korrespondenzen aufweisen.

Die Vorgehensweise zur Verlinkung von G_L ist im Algorithmus 5.3 dargestellt. Zuerst erfolgt die Ermittlung der Zusammenhangskomponenten von G_L bzw. seines Paargraphen G_P (Zeile 3). Anschließend werden Paare, die mit Kanten zwischen den Zusammenhangskomponenten korrespondieren, erneut verifiziert (Zeile 5) und stabile zu G_L und G_I hinzugefügt (Zeilen 6 und 7).

Falls selbst nach der Verlinkung noch nicht zusammenhängende G_L oder G_P existieren, so kann unter Umständen keine vollständige Bildverknüpfung hergestellt werden. Die Suche nach weiteren Verknüpfungen würde an dieser Stelle ohne Zusatzinformationen mehr oder weniger einem Zufallsprozess entsprechen. Daher erfolgt die Suche nach weiteren Verknüpfungen erst nach der Blockkonstruktion, wenn zusätzliche Information vorliegt (siehe Abschnitt 5.6).

5.4.2 Verdichtung

Die Konstruktion verfolgte bis hierhin das Ziel, einen Verknüpfungsgraphen zu konstruieren, der eine vollständige Bildverknüpfung ermöglicht. Dazu wurden Paarknoten korrespondierend mit

Eingabe: Bildgraph G_I, Verknüpfungsgraph G_L, Paare $\mathcal{P}$, Bildauflösung s

1 **procedure** `Verlinkung`$(G_I,\ G_L,\ \mathcal{P},\ s)$
2 **foreach** $G \in \{G_L, G_P\}$ **do** *Paargraph $G_P \subset G_L$*
3 $G_C \leftarrow$ `Komponenten`(G) *siehe Abschnitt 2.2.2*
4 $E = \left\{(u,v) \;\middle|\; \forall i \neq j :\ u \in V_C^i \wedge v \in V_C^j\right\}$ $G_C^i = (V_C^i, E_C^i) \in G_C$
5 $\mathcal{P}^* \leftarrow$ `Verifikation`$(\mathcal{P} \cap \mathcal{P}(E),\ s,\ G_L,\ G_I)$ *siehe Abschnitt 5.2*
6 $G_L \leftarrow G_L \oplus \mathcal{P}^*$ $\oplus$ *Grapherweiterung*
7 $G_I \leftarrow G_I \oplus \mathcal{P}^*$
8 **end**
9 **end**

Algorithmus 5.3: Verlinkung des Verknüpfungsgraphen

stabilen Paaren (Knotenpaare) zu G_L hinzugefügt. Die Herstellung von Verbindungen zwischen Paarknoten erfolgte über Kanten, wenn korrespondierende Knotenpaare gültig waren. Dabei kann es vorkommen, dass Kantenpaare eine bessere Qualität als inzidente Knotenpaare aufweisen. Über das Hinzufügen dieser Kantenpaare als Paarknoten lässt sich die Dichte von G_L erhöhen, was während der Blockkonstruktion (Abschnitt 5.5) zur Ermittlung von besseren Bildverknüpfungen führen kann.

Eingabe: Bildgraph G_I, Verknüpfungsgraph $G_L = (V_L, E_L)$, Bildauflösung s

1 **procedure** `Verdichtung`$(G_I,\ G_L,\ s)$
2 **do**
3 $E = \left\{e = (u,v) \in E_L \;\middle|\; \mathrm{Q}_{\mathcal{P}(e)} \geq \mathrm{Q}_{\mathcal{P}(u)} \wedge \mathrm{Q}_{\mathcal{P}(e)} \geq \mathrm{Q}_{\mathcal{P}(v)}\right\}$
4 $G_L \leftarrow G_L \oplus \mathcal{P}(E)$ $\oplus$ *Grapherweiterung*
5 $\mathcal{P}^* \leftarrow$ `Verifikation`$(\mathcal{P}(G_L),\ s,\ G_L,\ G_I)$ *siehe Abschnitt 5.2*
6 **while** $\mathcal{P}^* \neq \emptyset$
7 **end**

Algorithmus 5.4: Erhöhung der Dichte des Verknüpfungsgraphen

Die Erhöhung der Dichte des Verknüpfungsgraphen basiert auf dem Algorithmus 5.4. Darin werden iterativ neue Kanten in G_L ermittelt, deren korrespondierenden Paare eine mindestens so gute Qualität aufweisen wie Paare korrespondierend mit den inzidenten Paarknoten (Zeile 3). Die Qualität Q_P eines Paares P ist hierbei durch Gleichung (4.4) auf S. 58 definiert. Über das Hinzufügen korrespondierender, stabiler Kantenpaare aus $\mathcal{P}(E)$ zu G_L erfolgt schließlich die Erhöhung seiner Dichte (Zeilen 4 und 5).

5.5 Blockkonstruktion

Basierend auf dem konstruierten Verknüpfungsgraphen G_L erfolgt anhand Algorithmus 5.5 die Blockkonstruktion, d. h. die Bestimmung einer minimalen Teilmenge an Triplets für die

Bildverknüpfung. Das Ergebnis ist die Menge $\mathcal{B}$ von konstruierten Blöcken, die im Optimalfall lediglich einen vollständigen Block beinhaltet.

Eingabe: Verknüpfungsgraph G_L
Ausgabe: Blockmenge $\mathcal{B}$

```
1  B = ∅
2  G^C ← Komponenten(G_P)                          G_P ⊂ G_L, siehe Abschnitt 2.2.2
3  foreach G_P^C ∈ G^C do                                          G_L^C ⊃ G_P^C
4      G^C ← G^C \ G_P^C
5      G_L^S ← MinimalerTerminalerSteinerbaum(G_L^C)   G_L^S ⊆ G_L^C, siehe Abschnitt 2.2.4
6      T_S^-, T_S^× ← Verifikation(T(G_L^S), s, G_L^C, (∅,∅))        siehe Abschnitt 5.2
7      if (T_S^- ∪ T_S^×) = ∅ then
8          B ← B ∪ G_L^S
9      else
10         G^C ← G^C ∪ Komponenten(G_P^C)
11     end
12 end
```

Algorithmus 5.5: Blockkonstruktion

Als Erstes werden Zusammenhangskomponenten des Paargraphen $G_P \subset G_L$ zur Warteschlange G^C hinzugefügt (Zeile 2). Anschließend wird versucht, aus jedem Verknüpfungsteilgraphen $G_L^C \supset G_P^C$ in G^C einen Block zu bestimmen (Zeilen 3 bis 12). Dies geschieht über die Konstruktion des minimalen terminalen Steinerbaumes G_L^S (Zeile 5), gefolgt von der Verifikation der beinhalteten Triplets $\mathcal{T}(G_L^S)$ (Zeile 6). Die Operation $\mathcal{T}(G)$ liefert hierbei eine Menge $\mathcal{T}_G$ von Triplets korrespondierend mit den Kanten des Graphen G.

Falls keine ungültigen $\mathcal{T}_S^-$ und instabilen $\mathcal{T}_S^\times$ Triplets existieren, wird der Block G_L^S zur Blockmenge hinzugefügt (Zeile 8). Ansonsten werden diese Triplets aus dem Verknüpfungsteilgraphen entfernt (Zeile 6). Dies kann zum Zerfall von G_P^C in mehrere Zusammenhangskomponenten führen, für die dann jeweils ein Block zu bestimmen ist (Zeile 10).

Ungültige und instabile Triplets können zur Konstruktion unvollständiger Blöcke führen. Da korrespondierende Kanten im Verknüpfungsgraphen G_L entfernt wurden, könnte eine neue Konstruktion bzw. die Vervollständigung von G_L mittels Algorithmus 5.2, gefolgt von erneuter Blockkonstruktion zu größeren Blöcken führen. Falls die konstruierten Blöcke jedoch alle Bilder aus der Bildmenge beinhalten, ist es effizienter, die notwendigen Verknüpfungen direkt zwischen den Blöcken zu suchen (siehe Abschnitt 5.6).

5.6 Verknüpfung unvollständiger Blöcke

Die Blockkonstruktion (Abschnitt 5.5) liefert unter Umständen mehrere Blöcke. Diese können zur gleichen Szene gehören, die Verknüpfungen zwischen ihnen konnten jedoch mit den zur Verfügung stehenden Informationen nicht effizient ermittelt werden. Mittels hierarchischer Vereinigung (Abschnitt 5.7) können nun in jedem Block die relativen Kameraposen geschätzt werden, sodass

diese Zusatzinformation für die Suche nach fehlenden Verknüpfungen zwischen den Blöcken genutzt werden kann.

Die Gründe für das Fehlen der Verknüpfungen liegen im Wesentlichen in den großen Bildverzerrungen im Zusammenhang mit den eingesetzten Verfahren für die Kameraposeschätzung:

(1) Die Schätzung der Bildähnlichkeiten basiert auf einem schnellen Zuordnungsverfahren (Abschnitt 4.1), das nur begrenzt robust gegenüber größeren Bildverzerrungen ist. Das kann zu niedriger Gewichtung essenzieller Paare und folglich zur Nichtberücksichtigung bei der Konstruktion des Verknüpfungsgraphen (Abschnitt 5.4) führen.

(2) Die zur Verifikation von Paaren und Triplets (Abschnitt 5.2) benutzte Bildauflösung kann unzureichend sein. Dadurch können überlappende Bilder mit großen Verzerrungen einander nicht zugeordnet werden. Dies führt zu ungültigen Paaren bzw. Triplets und wenn diese essenziell für die Bildverknüpfung sind, zu unvollständigen Blöcken.

Für den ersten Grund gibt es keine Abhilfe, da alle Bilder miteinander verglichen werden und man somit auf ein schnelles, dafür aber weniger robustes Zuordnungsverfahren angewiesen ist. Um dem zweiten Grund entgegen zu wirken, kann die verwendete Auflösung erhöht werden. Das ist aber mit einem erheblich höheren Rechenaufwand verbunden. Da man nun aber über Zusatzwissen in Form von Kameraposen innerhalb eines Blocks verfügt, kann die Zuordnung mit einer höheren Auflösung auf eine kleine Menge von Paaren beschränkt werden.

Die Verknüpfung von zwei Blöcken B_1 und B_2 erfolgt über ein gemeinsames Paar P_{12}, das zur Übertragung der geometrischen Transformation dient. P_{12} kann dabei in folgenden Konstellationen vorkommen:

- *Verknüpfung über ein Paar*: Blöcke enthalten Triplets $T_1 = \{a, b, c\} \in B_1$ und $T_2 = \{a, b, d\} \in B_2$, die über zwei gemeinsame Bilder $T_1 \cap T_2 = \{a, b\} \in \mathcal{P}^*$ verfügen.
- *Verknüpfung über zwei Bilder*: Blöcke enthalten zwei gemeinsame Bilder $B_1 \cap B_2 = \{a, b\} \notin \mathcal{P}^\times$, die nicht notwendigerweise ein gültiges Paar bilden müssen.

Die Verknüpfung über ein Paar ist durch den Verknüpfungsgraphen beschrieben. Dieser ist jedoch nicht in der Lage Verknüpfung über zwei Bilder, die kein gültiges Paar bilden, zu modellieren. Nun können aber auch solche Verknüpfungen benutzt werden, um Blöcke zu verbinden. Die Verknüpfung über zwei beliebige Bilder hat den Vorteil, dass die verknüpfenden Bilder sich nicht überlappen müssen und somit größere Basen aufweisen können. Dadurch lässt sich die Gefahr von kritischen Konfigurationen reduzieren.

Aufgrund der höheren Robustheit und Flexibilität wird die Verknüpfung über zwei Bilder angestrebt. Diese kann zur Verknüpfung über ein Paar führen, wenn sich die Blöcke nur an einer Stelle überlappen und die Verknüpfung somit ausschließlich über ein Paar möglich ist.

Die Suche nach Verknüpfungen zwischen zwei Blöcken G_L^1 und G_L^2 ist im Algorithmus 5.6 zusammengefasst. Sie basiert auf dem Verknüpfungsgraphen G_L^{12}, der sich aus den Verknüpfungsgraphen der Blöcke durch ihre Verschmelzung (Symbol $\circledast$) zusammensetzt (Zeile 2). Ein zusammenhängender G_L^{12} ist eine notwendige Bedingung für die Verknüpfung der Blöcke und sichert die Verknüpfung über mindestens ein Bild zu. Für die Übertragung des Maßstabs sind jedoch zwei

Eingabe: Blöcke $G_L^1 = (V_L^1, E_L^1)$ und $G_L^2 = (V_L^2, E_L^2)$, Bildgraph G_I
Ausgabe: Block B

1 $B = (\emptyset, \emptyset)$ *leerer Graph*
2 $G_L^{12} \leftarrow G_L^1 \circledast G_L^2$ $\circledast$ *Graphverschmelzung*
3 $\mathcal{P}_{12} = \{(b_1, b_2) \mid b_1 \widehat{=} v_1 \in V_L^1 \wedge b_2 \widehat{=} v_2 \in V_L^2\}$ *gemeinsame Paare*
4 $\mathcal{P}_{12} \leftarrow \mathcal{P}_{12} \backslash \texttt{Ausreißer}(\mathcal{P}_{12})$
5 $G_L^{12} \leftarrow G_L^{12} \oplus \mathcal{P}_{12}$ $\oplus$ *Grapherweiterung (siehe Abschnitt 5.4)*
6 $\texttt{Verifikation}(\mathcal{P}(G_L^{12}),\ s_p,\ G_L^{12},\ G_I)$ *siehe Abschnitt 5.2*
7 $\texttt{Verifikation}(\mathcal{T}(G_L^{12}),\ s_t,\ G_L^{12},\ (\emptyset, \emptyset))$
8 **if** $|\texttt{Komponenten}(G_L^{12})| = 1$ **then**
9 $\quad \mathcal{P}_P = \{P \mid P \subseteq (\mathcal{P}(G_L^{12}) \cap \mathcal{P}_{12}) \wedge 1 \leq |P| \leq 2\}$
10 $\quad$ wähle $P \in \mathcal{P}_P$ sodass $\left(\forall P^i \in P :\ G_L^{i'} \leftarrow G_L^i \oplus P^i\right) \rightarrow |V_L^{1'} \cap V_L^{2'}| \geq 2$
11 $\quad$ **if** $P \neq \emptyset$ **then**
12 $\quad\quad \forall P^i \in P :\ G_L^i \leftarrow G_L^i \oplus P^i$
13 $\quad\quad B \leftarrow G_L^1 \circledast G_L^2$
14 $\quad$ **end**
15 **end**

Algorithmus 5.6: Verlinkung von zwei Blöcken

Bilder erforderlich. Eine hinreichende Bedingung ist hingegen ein zusammenhängender Paargraph $G_P^{12} \subset G_L^{12}$, der eine Verknüpfung über ein Paar ermöglicht.

Um fehlende Verknüpfungen zu finden, werden *gemeinsame Paare* $\mathcal{P}_{12}$, d. h. Paare deren Bilder sich in verschiedenen Blöcken befinden, ermittelt (Zeile 3). Unwahrscheinliche Paare werden, basierend auf den geschätzten Bildähnlichkeiten, mittels der X84-Regel (Abschnitt 2.3) detektiert und aus $\mathcal{P}_{12}$ entfernt (Zeile 4). Das Hinzufügen der verbleibenden zu G_L^{12} führt zu Paaren $\mathcal{P}(G_L^{12})$ und Triplets $\mathcal{T}(G_L^{12})$, die mit den resultierenden Knoten und Kanten korrespondieren (Zeile 5). Im Rahmen der Verifikation erfolgt die Entfernung von nicht stabilen Paaren und Triplets aus G_L^{12} (Zeilen 6 und 7).

Falls die notwendige Bedingung für die Existenz von Verknüpfungen erfüllt ist (Zeile 8) erfolgt die Ermittlung von geeigneten Paaren P, sodass die Blöcke verknüpft werden können (Zeilen 9 und 10). Dabei wird nach einem oder zwei Paaren gesucht, die Bilder aus einem Block zum anderen hinzufügen können. Damit würden die Blöcke zwei gemeinsame Bilder beinhalten, über welche die Verknüpfung erfolgen kann. War die Suche erfolgreich, werden die ermittelten Paare zu den Blöcken hinzugefügt und die Blöcke über die gemeinsamen Bilder zum Block B verknüpft (Zeilen 11 bis 14).

5.7 Vereinigung von Bildteilmengen

Basierend auf den hergestellten Bildverknüpfungen (Abschnitte 5.4 und 5.5) erfolgt über die hierarchische Vereinigung von Bildteilmengen nach (Mayer 2014) die Schätzung von relativen Kameraposen (vgl. Abschnitt 2.1.8). Das primäre Ziel in (Mayer 2014) bestand in der Punktreduktion zur Effizienzsteigerung. Dieses Kapitel befasst sich hingegen mit einer Optimierungsstrategie zur Verbesserung der Effizienz durch bessere Ausnutzung der Parallelität sowie Erhöhung der Robustheit der Vereinigung.

Eine *Bildteilmenge* (BTM) bestehend aus $n = |\text{BTM}|$ Bildern wird als n-BTM bezeichnet. Die hierarchische Vereinigung startet mit Triplets als 3-BTM und vereinigt diese zu 4-BTM. Im Allgemeinen erfolgt die Vereinigung zweier Bildteilmengen BTM_i und BTM_j mit k gemeinsamen Bildern zu einer $(|\text{BTM}_i| + |\text{BTM}_j| - k)$-BTM. Die Transformation ins gemeinsame Koordinatensystem erfolgt über eine (starre) euklidische Transformation, die anhand von zwei gemeinsamen Bildern bestimmt wird. Somit ist $k \geq 2$ eine *notwendige Bedingung* für die Vereinigung zweier Bildteilmengen. Zudem sollen die gemeinsamen Bilder eine ausreichende Basislänge aufweisen, um die Bestimmung eines zuverlässigen Maßstabs sicherzustellen.

Nachdem sich die Bildteilmengen im gemeinsamen Koordinatensystem befinden, erfolgt die Übertragung der 3D-Punkte, gefolgt von der robusten Bündelausgleichung (Abschnitt 2.1.3). Die Genauigkeit der 3D-Punkte hängt hierbei von der Anzahl an Beobachtungen ab, d. h. von der Anzahl an Bildern in denen die 3D-Punkte sichtbar sind (Triggs et al. 1999). Größere Bildteilmengen enthalten tendenziell genauere 3D-Punkte als kleinere, weil die 3D-Punkte in der größeren Bildteilmenge in mehr Bildern sichtbar sind. Die Anzahl an Beobachtungen pro 3D-Punkt ist jedoch, beispielsweise aufgrund von Verdeckungen, vor allem aber wegen der räumlichen Verteilung und unterschiedlichen Orientierungen der Kameras in größeren Bildmengen, meist auf einige wenige Bilder begrenzt.

Im Rahmen der robusten Bündelausgleichung erfolgt die Gewichtung der 3D-Punkte basierend auf ihrer Genauigkeit sowie die Eliminierung von unzureichend genauen 3D-Punkten. Letzteres kann bei der Vereinigung von zwei Bildteilmengen mit stark abweichender Größe zum Verlust essenzieller 3D-Punkte und somit instabiler Vereinigung führen, weil die Punkte in der kleineren Bildteilmenge ungenauer sind und damit leichter eliminiert werden. Hieraus ergibt sich die *Anforderung*, dass die zu vereinigende Bildteilmengen, bis zu einer gewissen Größe, eine ähnliche Größe aufweisen sollen.

5.7.1 Vereinigungsregeln

Die Vereinigung von Bildteilmengen wird über sogenannte *Vereinigungsregeln* (kurz: Regeln) beschrieben. Diese besagen welche Bildteilmengen miteinander zu vereinigen sind und basierend auf welchen Bildern die geometrische Transformation zu bestimmen ist. Vereinigungsregeln lassen sich in mehrere Ebenen unterteilen. Alle Regeln, die sich auf der gleichen Ebene befinden, sind unabhängig voneinander. Lediglich zwischen Regeln auf unterschiedlichen Ebenen können Abhängigkeiten existieren. Regeln auf der Ebene $i+1$ können erst verarbeitet werden, wenn die Regeln auf der Ebene i verarbeitet wurden. Die Regeln für die Vereinigung von Triplets befinden sich auf der niedrigsten Ebene und für die Vereinigung der letzten beiden Bildteilmengen auf der

höchsten. Auf diese Weise wird ein sogenannter *Regelbaum* definiert (siehe Abb. 5.2), der die Vereinigungsreihenfolge und die Abhängigkeiten zwischen den Regeln beschreibt.

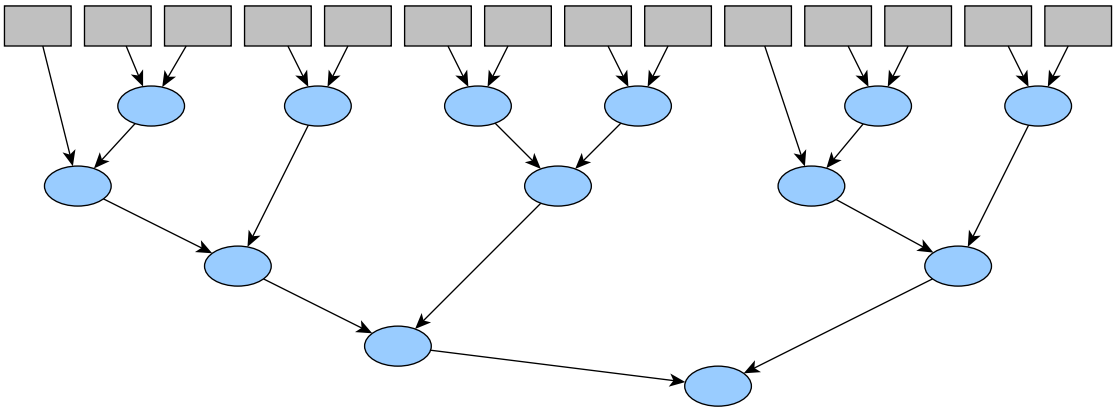


Abbildung 5.2: Regelbaum mit 6 Ebenen und 13 Regelknoten (blaue Ellipsen) für die Vereinigung von 14 Triplets (graue Rechtecke). Jedes Triplet entspricht einer 3-BTM und jeder Regelknoten einer Regel. Die Pfeile geben die Vereinigungsrichtung an.

5.7.2 Generierung von Vereinigungsregeln

Ein Vorteil der hierarchischen Vereinigung besteht darin, dass die einzelnen Vereinigungen unabhängig voneinander sind und somit parallel verarbeitet werden können. Um dies unter Beachtung der Anforderung einer ähnlichen Größe möglichst gut ausnutzen zu können, muss die Reihenfolge der Vereinigungen beachtet werden. Die Vereinigung in (Mayer 2014) entspricht einer agglomerativen, hierarchischen Clusterbildung (vgl. Abschnitt 2.6) mit Bildteilmengen als Cluster. Sie startet mit Triplets als Cluster bestehend aus drei Bildern und bildet immer größere Cluster durch die Minimierung ihrer Distanz

$$d_c\left(\mathrm{BTM}_i, \mathrm{BTM}_j\right) = \min\left(|\mathrm{BTM}_i|, |\mathrm{BTM}_j|\right) \ . \tag{5.6}$$

Der Vorteil dieser Vorgehensweise liegt in der effizienten Generierung von Vereinigungsregeln. Allerdings erfolgt keine Berücksichtigung des Größenunterschieds zwischen den Bildteilmengen. Darüber hinaus führen die resultierenden Vereinigungsregeln tendenziell zu einer suboptimalen Parallelisierung. Dies wird deutlich, wenn ihre Generierung als Paarung im Bildteilmengengraphen formuliert wird.

Ein *Bildteilmengengraph* ist ein gewichteter Graph $G_B = (V_B, E_B)$, der die Beziehungen zwischen den vereinigbaren Bildteilmengen beschreibt. Seine Knoten korrespondieren mit den Bildteilmengen und eine Kante existiert nur, wenn die inzidenten Knoten die notwendige Bedingung für die Vereinigung erfüllen (d. h. zwei gemeinsame Bilder besitzen). Die Generierung von Vereinigungsregeln lässt sich als Paarungsproblem in G_B formulieren (siehe Abschnitt 2.2.6). Jede Paarung führt zum Minor[1] des Bildteilmengengraphen und der Generierung von Regeln einer

[1] Ein *Minor* ist ein Graph, der durch Kantenkontraktion aus einem anderen Graphen entstanden ist. Dabei versteht man unter *Kantenkontraktion* die Entfernung einer Kante und die Vereinigung der inzidenten Knoten zu einem neuen Knoten.

Ebene im Regelbaum. Iterative Suche nach Paarungen in den resultierenden Minoren, gefolgt von der Bildung neuer, führt zu immer kleineren Bildteilmengengraphen und somit zum Wachstum des Regelbaumes.

Um den höchstmöglichen Parallelisierungsgrad zu erzielen, ist eine perfekte Paarung erforderlich. Diese erfordert jedoch besondere Grapheigenschaften (vgl. Abschnitt 2.2.6), die bei G_B nicht immer gegeben sind. Aus diesem Grund wird die Generierung der Vereinigungsregeln als die Suche nach einer maximalen Paarung formuliert. Häufig entspricht eine maximale Paarung jedoch einer perfekten und man erhält Regeln, die in diesem Zusammenhang hinsichtlich der Parallelisierung optimal sind.

Die clusterbasierte Vereinigung in (Mayer 2014) entspricht einer nicht erweiterbaren Paarung in G_B und kann somit suboptimale Regeln hinsichtlich der Parallelisierung liefern. Eine maximale Paarung ist jedoch immer nicht erweiterbar, sodass diese zu Regeln führt, die hinsichtlich der Parallelisierung mindestens so gut sind wie die Regeln, die mittels einer clusterbasierten Generierung erzeugt werden können. Das liegt daran, dass mittels maximaler Paarung mehr Regeln pro Ebene erzeugbar sind, was zum höheren Parallelisierungsgrad führt (siehe Abschnitt 6.5).

Über die Gewichtung der Kanten und die Suche nach maximaler Paarung von minimalem Gewicht lassen sich optimale Vereinigungsregeln hinsichtlich der Parallelisierung unter Beachtung der Anforderung ähnlicher Größen generieren.

Die Forderung nach der Vereinigung von ähnlich großen Bildteilmengen ist hierbei in zweierlei Hinsicht vorteilhaft. Zum einen führt dies zu stabilerer Vereinigung aufgrund von ähnlichen Punktgenauigkeiten. Zum anderen wird hierdurch die Degeneration zu sequenzieller Vereinigung (vgl. Abschnitt 2.1.8) vermieden. Ein stetiges Wachstum der Bildteilmengen ließe sich über die Gewichtsfunktion

$$w_k(e) = d_c\left(\mathrm{BTM}_i, \mathrm{BTM}_j\right) \tag{5.7}$$

erzielen, wobei $e \in E_B$ einer Kante zwischen Knoten korrespondierend mit den Bildteilmengen BTM_i und BTM_j entspricht. Auf diese Weise würden kleinere Bildteilmengen bevorzugt vereinigt werden, allerdings unabhängig vom Größenunterschied. Somit kann beispielsweise zwischen der Vereinigung einer n-BTM mit einer $(n+1)$-BTM oder einer n^2-BTM nicht unterschieden werden.

Die Einbeziehung des Größenunterschieds zwischen den zu vereinigenden Bildteilmengen kann über die Gewichtsfunktion

$$w_d(e) = \left||\mathrm{BTM}_i| - |\mathrm{BTM}_j|\right| \tag{5.8}$$

erreicht werden. Diese stellt sicher, dass zuerst immer möglichst ähnliche Bildteilmengen vereinigt werden. Allerdings hätte hierdurch die Vereinigung nicht vereinigter Bildteilmengen auf der n-ten Ebene eine niedrigere Priorität für die Vereinigung auf der $(n+1)$-ten Ebene. Das würde wiederum zu immer stärker abweichendem Wachstum der Bildteilmengen und somit einem entarteten Regelbaum führen. Letztendlich müssten dann irgendwann Bildteilmengen mit stark abweichenden Größen vereinigt werden.

Um dies zu vermeiden, sind die Vereinigungsregeln so zu generieren, dass möglichst jede Bildteilmenge in jeder Ebenen vereinigt wird. Auf diese Weise würden die resultierenden Bildteilmengen immer gleich schnell wachsen. Da dies nicht immer möglich ist, werden stattdessen Bildteilmengen,

die in der aktuellen Ebenen nicht vereinigt werden konnten, in der nächsten Ebene bevorzugt. Dazu wird die Gewichtsfunktion

$$w_c(e) = \mathrm{e}^{\max(c_i, c_j)} \tag{5.9}$$

eingesetzt, die ein Gewicht in Abhängigkeit von der maximalen Anzahl c_k an Ebenen, in denen die Bildteilmenge BTM_k nicht vereinigt wurde, zuweist. Dadurch hängt die Priorität der Vereinigung einer Bildteilmenge von der Anzahl an Ebenen ab, in denen diese nicht vereinigt wurde.

Die Kombination der Gewichtsfunktionen w_d und w_c führt schließlich zur Gewichtsfunktion

$$w_{dc}(e) = w_d(e)\, w_c(e) \tag{5.10}$$

des Bildteilmengengraphen, die eine Vereinigung von ähnlich großen Bildteilmengen bei stetigem Wachstum forciert.

5.8 Filterung instabiler Konfigurationen

Im Falle nicht erfolgreicher hierarchischer Vereinigung, enthält der entsprechende Block Verknüpfungen mit schwachen, jedoch nicht notwendigerweise instabilen, Konfigurationen. Um diese zu ermitteln, wird der Pfad im Regelbaum (vgl. Abschnitt 5.7.1) bis zur gescheiterten Regel betrachtet. Diente die gescheiterte Regel zur Vereinigung größerer Bildteilmengen, so wurde wahrscheinlich ein Paar mit unzureichender Basis zur Bestimmung des Maßstabs benutzt. Allerdings kann auch an einer anderen Stelle im Regelbaum ein Paar oder Triplet verwendet worden sein, das zur Deformationen führte, die sich erst später bemerkbar machten.

Deswegen werden alle Paare und Triplets, die im Pfad enthalten sind, betrachtet. Die Paare mit der niedrigsten Qualität Q_P (vgl. Abschnitt 4.4.2) werden als instabil betrachtet und aus dem Verknüpfungsgraphen entfernt. Die Ermittlung schwacher Triplets erfolgt basierend auf der Anzahl an rekonstruierten 3D-Punkten mittels der X84-Regel (Abschnitt 2.3). Diese werden als ungültig eingestuft und die korrespondierenden Kanten aus dem Verknüpfungsgraphen entfernt.

Anschließend wird wie in Abb. 5.1 auf S. 69 dargestellt der Verknüpfungsgraph erneut konstruiert bzw. vervollständigt, um fehlende Bildverknüpfungen für die nachfolgende Blockkonstruktion wieder herzustellen.

6 Experimente

Dieses Kapitel befasst sich auf Grundlage praktischer Experimente mit der Analyse der Konzepte, die in den vorangegangenen Kapiteln vorgestellt wurden. Diese erfolgten auf dem System, dessen Spezifikation in Tab. 6.1 aufgelistet ist, anhand den im Abschnitt 6.1 beschriebenen Bildmengen.

Prozessor (CPU)	2 × Intel® Xeon® E5-2643 v3 (6 Kerne, 3,40 GHz)
Grafikkarte (GPU)	NVIDIA GeForce GTX Titan Z (5760 CUDA-Einheiten, 705 MHz)
Arbeitsspeicher	256 GB
Betriebssystem	Windows 10 x64
Compiler	Visual Studio 2015

Tabelle 6.1: Spezifikation des Evaluierungssystems

Die Auswirkungen der Deskriptoreinbettung auf die Bildähnlichkeiten werden im Abschnitt 6.2 untersucht. Mit der Detektion von kritischen Kamerakonfiguration mittels Klassifikation befasst sich Abschnitt 6.3. Im Abschnitt 6.4 werden Strategien für den Schleifenschluss und im Abschnitt 6.5 die Vereinigung von Bildteilmengen analysiert. Der Vergleich des entwickelten Verfahrens mit einigen existierenden Verfahren erfolgt im Abschnitt 6.6. Schließlich werden die erzielten Ergebnisse im Abschnitt 6.7 zusammengefasst und bewertet.

6.1 Bildmengen

Die Experimente nutzen Bildmengen, deren Eigenschaften in Tabelle 6.2 aufgelistet sind. Die mit dem Verfahren aus dieser Arbeit geschätzten Kameraposen der einzelnen Bildmengen sind in Abb. 6.1 bis 6.4 dargestellt. Die Pyramiden repräsentieren Kameraposen, wobei die Pyramidenspitze der Kameraposition und die Orientierung der Pyramide der Kameraorientierung entspricht. Kameratypen sind durch Pyramiden gleicher Farbe gekennzeichnet.

Bei den Aufnahmen wird zwischen Boden-, Drohnen- und Luftaufnahmen unterschieden. *Bodenaufnahmen* wurden mit einer handelsüblichen Kamera vom Boden aus aufgenommen. Bei den *Drohnenaufnahmen* stammen die Aufnahmen von einer Kamera, die auf eine Drohne (engl. *unmanned aerial vehicle – UAV*) montiert wurde. Die *Luftaufnahmen* wurden mit einer speziellen, hochauflösenden Luftbildkamera erstellt. Diese befindet sich in einem Flugzeug, das meistens auf einer relativ großen Höhe fliegt.

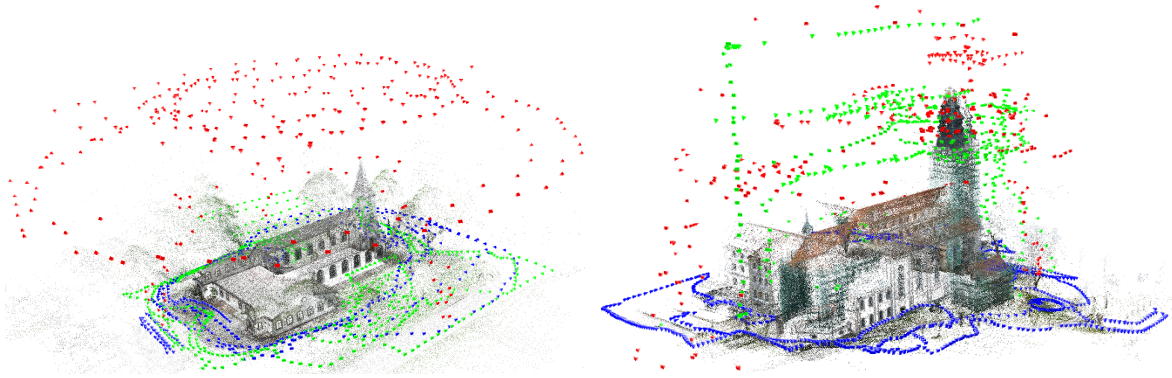
Die Luftbildaufnahmen weisen sehr häufig eine vordefinierte Aufnahmekonfiguration mit festgelegten Überlappungsbereichen auf. Somit enthalten sie in der Regel keine kritischen Konfigurationen.

Bildmenge	Größe	Kameras	Boden-aufnahmen	Drohnen-aufnahmen	Luft-aufnahmen
Dorf-Überflug	232	1		×	
Piazza Bra (Toldo et al. 2015)	331	1	×		
UQ St Lucia (Warren et al. 2010)	351	1	×		
Kirche	1455	3	×	×	
Kloster	1769	3	×	×	
Übungsplatz	3664	1		×	
Flugplatz	6210	2		×	
Dorf	6405	10	×	×	×

Tabelle 6.2: Eigenschaften der Bildmengen

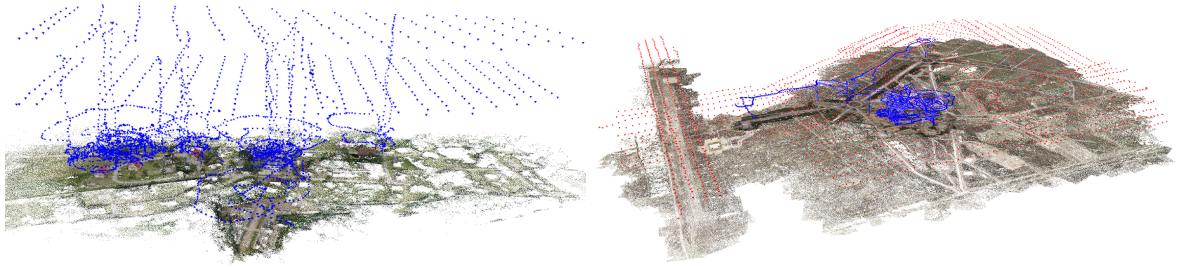
Im Gegensatz dazu können bei Boden- und Drohnenaufnahmen beliebige Aufnahmekonfigurationen vorkommen, die sowohl zu großen relativen Bildverzerrungen als auch kritischen Konfigurationen führen können. So entstehen beispielsweise bei zeitgesteuerter Aufnahme mit einer ruhenden oder ausschließlich gedrehten Drohne Bewegungsdegenerationen (vgl. Abschnitt 2.1.6) zwischen den Aufnahmen. Selbst wenn die Aufnahmekonfigurationen in den einzelnen Fällen vordefiniert sind, kann ihre Kombinationen zu komplexen Bildmengen (vgl. Kapitel 1) führen.

Bei der Bildmenge *UQ St Lucia* handelt es sich um eine Teilmenge des Datensatzes von Warren et al. (2010). Dieser besteht aus 66394 Bildern aufgenommen mit einem Stereokamerasystem, das auf einem Fahrzeug montiert wurde. Die benutze Bildmenge enthält die Aufnahmen einer Kamera im Abstand von etwa einem Meter und repräsentiert eine Bildschleife (siehe Abb. 6.10(a)). Für die Experimente mit dieser Bildmenge wird die mitgelieferte Kalibrierung verwendet.



(a) Kirche (b) Kloster

Abbildung 6.1: Bildmengen *Kirche* (1455 Bilder, 3 Kameras) und *Kloster* (1769 Bilder, 3 Kameras), die sich aus Boden- und Drohnenaufnahmen zusammensetzen.



(a) Übungsplatz (b) Flugplatz

Abbildung 6.2: Bildmengen *Übungsplatz* (3664 Bilder, eine Kamera) und *Flugplatz* (6210 Bilder, 2 Kameras), die sich ausschließlich aus Drohnenaufnahmen zusammensetzen.

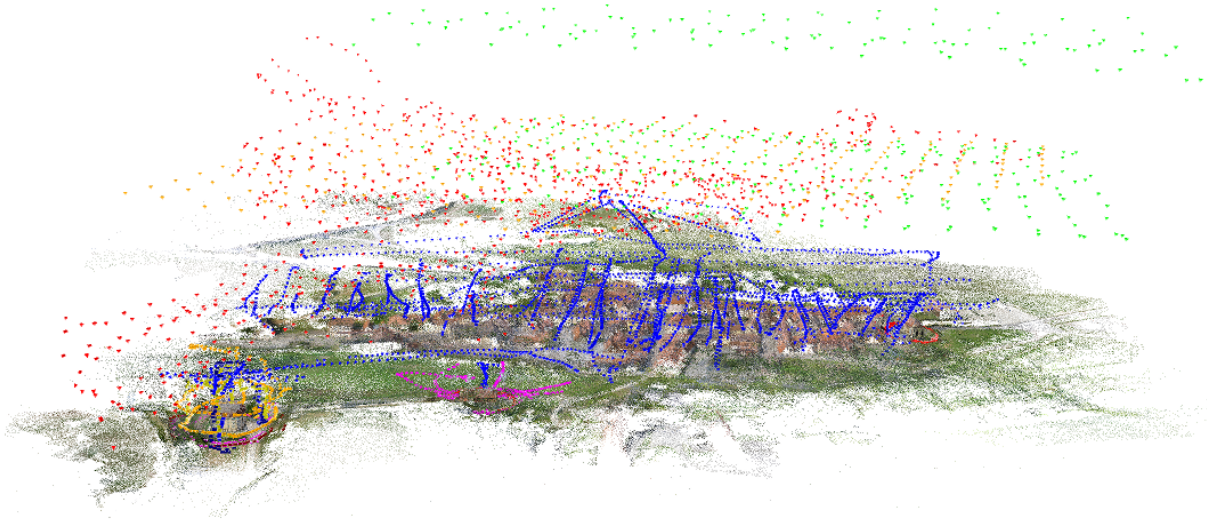


Abbildung 6.3: Bildmenge *Dorf* bestehend aus 6405 Boden-, Drohnen- und Luftbildaufnahmen von 10 unterschiedlichen Kameras.

6.2 Schätzung der Bildähnlichkeiten

Um Bildähnlichkeiten schneller bestimmen zu können, werden reelle SIFT-Deskriptoren auf binäre abgebildet (vgl. Abschnitt 4.1.1). Dies ermöglicht neben einer speichereffizienteren Repräsentation einen schnelleren Deskriptorvergleich mittels Bitoperationen (vgl. Abschnitt 4.1.2). In diesem Abschnitt erfolgt die Untersuchung der erzielbaren Effizienzsteigerung (Abschnitt 6.2.1) sowie der Auswirkung auf die ermittelten Bildähnlichkeiten (Abschnitt 6.2.2).

6.2.1 Laufzeit

In Abb. 6.5 sind Laufzeiten für die Korrespondenzsuche unter Verwendung von reellen Originaldeskriptoren sowie eingebetteten Deskriptoren dargestellt. Im Falle der Letzteren kommt zu der Laufzeit noch die Zeit für die Einbettung hinzu. Allerdings ist diese praktisch vernachlässigbar, da sie selbst für 6500 Bilder bei unter zwei Sekunden liegt. Für die Korrespondenzsuche zwischen

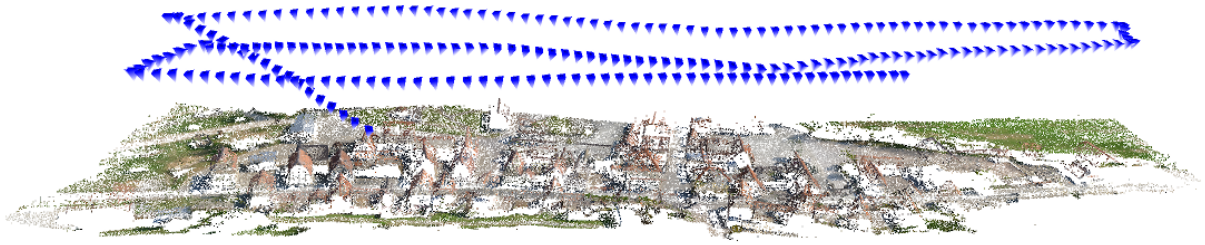


Abbildung 6.4: Bildmenge *Dorf-Überflug* als Teilmenge der Bildmenge *Dorf* bestehend aus 232 Aufnahmen einer Drohne.

Originaldeskriptoren wurde eine grafikkartenbasierte Implementierung (Wu 2012) benutzt. Der Deskriptorvergleich erfolgt hierbei über die Kosinusdistanz. Dies ermöglicht eine Formulierung als Matrixmultiplikation, die auf der Grafikkarte effizient durchgeführt werden kann. Der Vergleich von eingebetteten Deskriptoren wird hingegen nicht auf der Grafikkarte durchgeführt, sondern benutzt ausschließlich die Streaming SIMD Extensions (SSE) des Prozessors zur Beschleunigung der Bitoperationen.

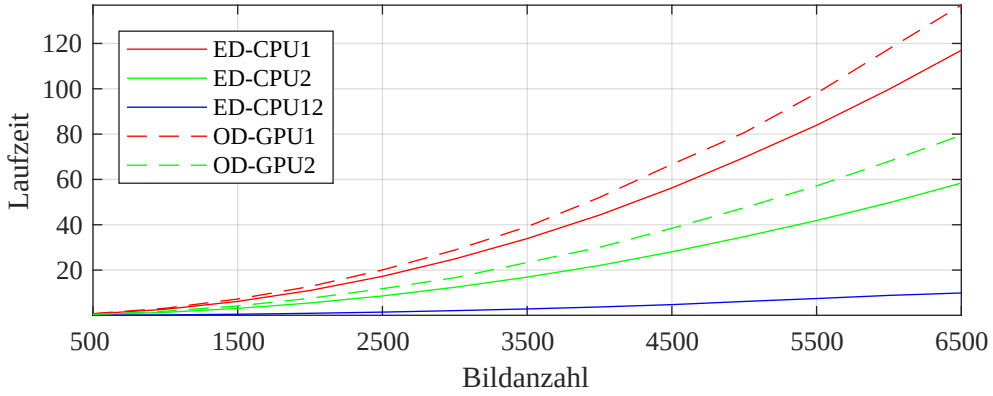


Abbildung 6.5: Laufzeiten in Minuten für die Korrespondenzsuche zwischen Originaldeskriptoren (OD) sowie eingebetteten Deskriptoren (ED). CPUn gibt die Anzahl n an verwendeten Prozessorkernen und GPUm die Anzahl m an benutzten Grafikkartenprozessoren an.

Die Korrespondenzsuche basierend auf den eingebetteten Deskriptoren weist deutlich bessere Laufzeiten auf. Selbst die Benutzung von nur einem Prozessorkern (CPU1) führt zu schnellerer Korrespondenzsuche im Vergleich zur Verwendung eines Grafikkartenprozessors (GPU1). Hierbei ist zu beachten, dass ein Grafikkartenprozessor mehrere hundert Ausführungseinheiten (Shader) besitzt und die Korrespondenzsuche parallel erfolgt.

6.2.2 Qualität

Die geschätzten Bildähnlichkeiten werden benutzt, um die Anzahl an aufwändigen robusten Kameraposeschätzungen zu reduzieren (vgl. Kapitel 4). Somit ist anzunehmen, dass die Qualität

der Bildähnlichkeiten einen direkten Einfluss auf die Anzahl an robusten Kameraposeschätzungen hat. Das bedeutet, je besser die Bildähnlichkeiten geschätzt wurden, desto weniger ungültige Paare sollten während der Blockkonstruktion auftreten. Auf Triplets lässt sich diese Schlussfolgerung allerdings nicht ohne Weiteres übertragen. Diese benötigen dreifache Überlappungen, die aus den paarweisen Bildähnlichkeiten nicht direkt ableitbar sind (vgl. Abschnitt 4.4.1).

Die Beurteilung der Qualität der Bildähnlichkeiten erfolgt daher über die Betrachtung der Anzahl an verifizierten Paaren während der Blockkonstruktion. Zudem werden die Anzahl verifizierter Triplets und der größte konstruierte Block betrachtet. Obwohl die beiden Letzteren nicht in direktem Zusammenhang mit den Bildähnlichkeiten stehen, wird davon ausgegangen, dass genauere Bildähnlichkeiten einen indirekten Einfluss auf die Bestimmung der Triplets und folglich auch die Blockgröße haben.

Die Ergebnisse der Blockkonstruktion unter Verwendung von Bildähnlichkeiten basierend auf den eingebetteten und den Originaldeskriptoren sind in Tabelle 6.3 aufgelistet. Um mögliche Nebeneinflüsse zu reduzieren, wurde die Detektion von kritischen Kamerakonfigurationen abgeschaltet und die Blockkonstruktion ausschließlich anhand der Anzahl an Korrespondenzen durchgeführt. Außerdem erfolgte keine Verknüpfung unvollständiger Blöcke (vgl. Abschnitt 5.6), sodass die Blockgröße ausschließlich von den Bildähnlichkeiten abhängt.

Bildmenge	$\frac{\lvert\mathcal{P}^-_{OD}\rvert}{\lvert\mathcal{P}_{OD}\rvert}$	$\frac{\lvert\mathcal{P}^-_{ED}\rvert}{\lvert\mathcal{P}_{ED}\rvert}$	$\frac{\lvert\mathcal{P}_{ED}\rvert}{\lvert\mathcal{P}_{OD}\rvert}$	$\frac{\lvert\mathcal{T}^-_{OD}\rvert}{\lvert\mathcal{T}_{OD}\rvert}$	$\frac{\lvert\mathcal{T}^-_{ED}\rvert}{\lvert\mathcal{T}_{ED}\rvert}$	$\frac{\lvert\mathcal{T}_{ED}\rvert}{\lvert\mathcal{T}_{OD}\rvert}$	$\frac{B_{OD}}{N}$	$\frac{B_{ED}}{N}$
Kirche	0,005	0,004	0,999	0,028	0,024	0,996	0,353	0,371
Übungsplatz	0,008	0,011	1,027	0,002	0,002	1,001	0,966	0,989
Flugplatz	0,007	0,010	1,009	0,003	0,003	0,980	0,409	0,478
Dorf	0,054	0,060	1,012	0,025	0,024	0,999	0,408	0,368

Tabelle 6.3: Vergleich der Blockkonstruktion unter Verwendung von eingebetteten (ED) sowie Originaldeskriptoren (OD) für die Schätzung der Bildähnlichkeiten. $|\mathcal{P}^-|$ bzw. $|\mathcal{T}^-|$ bezeichnet die Anzahl der ungültigen Paare bzw. Triplets, $|\mathcal{P}|$ bzw. $|\mathcal{T}|$ die Gesamtanzahl der verifizierten Paare bzw. Triplets, B die Anzahl der Bilder im größten konstruierten Block und N die Größe der jeweiligen Bildmenge.

Die Verwendung von eingebetteten Deskriptoren führt zu einer leicht erhöhten Anzahl an ungültigen Paaren. Dies spiegelt sich auch in der Gesamtanzahl an verifizierten Paaren wider. Interessanterweise hat es den Anschein, als ob Bildähnlichkeiten basierend auf eingebetteten Deskriptoren eine geeignetere Tripletauswahl ermöglichen, sodass in den meisten Fällen größere Blöcke konstruiert werden konnten.

Zusammenfassend lässt sich feststellen, dass die Verwendung von eingebetteten Deskriptoren im Vergleich zu Originaldeskriptoren die Effizienz deutlich erhöht sowie zu ähnlichen Ergebnissen hinsichtlich der Blockkonstruktion führt. Ein möglicher Grund hierfür könnte in der beschränkten Eignung des Ähnlichkeitsmaßes oder allgemein der Anzahl an Korrespondenzen für die Auswahl der Paare liegen. Zum einen garantiert eine hohe Korrespondenzanzahl keine ausreichende Überlappung und zum anderen spielen auch andere Faktoren, wie beispielsweise die Verteilung der korrespondierenden Punkte in den Bildern, eine wichtige Rolle. Somit haben

einbettungsbedingte Abweichungen keinen gravierenden Einfluss auf die Paarauswahl, solange die relativen Abweichungen der Bildähnlichkeiten weitgehend erhalten bleiben.

Ein weiterer Umstand, der zugunsten von eingebetteten Deskriptoren sprechen könnte ist, dass durch die Quantisierung Störeinflüsse in den Originaldeskriptoren eliminiert werden konnten, sodass diese aussagekräftiger und robuster hinsichtlich Bildverzerrungen sind. Ähnliche Beobachtungen wurden bereits in (Ke und Sukthankar 2004) und (Bay et al. 2006) bei der Approximation von SIFT gemacht.

6.3 Klassifizierung von Paaren

Die Klassifizierung von Paaren basiert auf einem Random Forest Klassifikator (vgl. Abschnitt 2.5.3) mit den im Abschnitt 5.1 definierten Merkmalen. Dieser Abschnitt befasst sich mit der Ermittlung optimaler Parameter für den Klassifikator sowie Merkmalskombinationen mit dem Ziel eines minimalen Klassifikationsfehlers.

6.3.1 Trainingsdaten

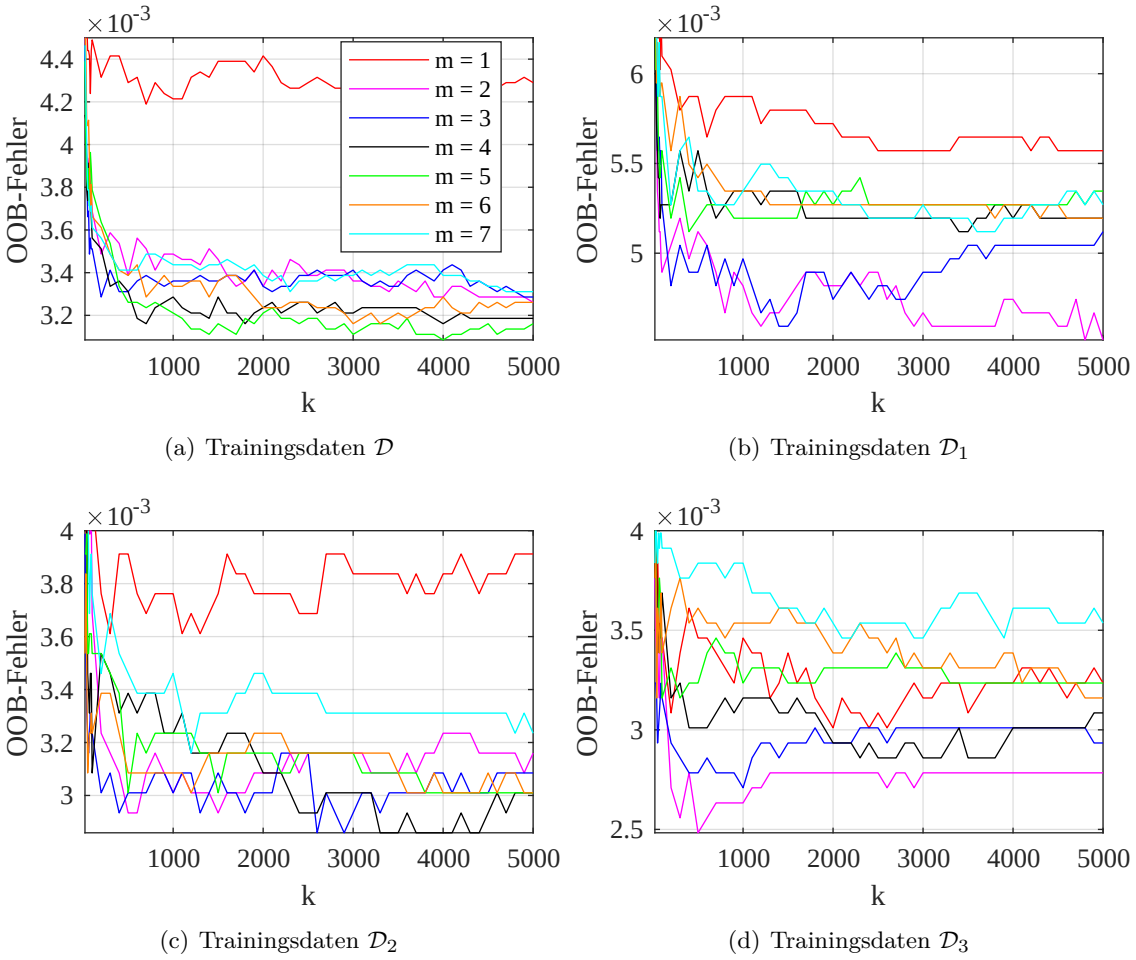
Die Trainingsdaten basieren auf 3492 nicht degenerierten und 3154 degenerierten Paaren, d. h. Paaren mit Bewegungsdegeneration (vgl. Abschnitt 2.1.6). Letztere bestehen aus Bodenaufnahmen mit reiner Rotation ohne Verschiebung. Nicht degenerierte Paare enthalten sowohl Boden- als auch Luft- und Drohnenaufnahmen bei denen sichergestellt wurde, dass eine Basis existiert. Die robuste Kameraposeschätzung nach (Mayer et al. 2012) erfolgte auf drei verschiedenen Bildauflösungen s_1, $s_2 = 2s_1$ und $s_3 = 4s_1$ unter Vertauschung des Referenzbildes (vgl. Abschnitt 5.2). Daraus ergaben sich drei Trainingsdatenmengen $\mathcal{D}_i$ für jede Bildauflösung s_i sowie eine zusammengesetzte $\mathcal{D} = \bigcup \mathcal{D}_i$ bestehend aus 18924 degenerierten und 20952 nicht degenerierten Paaren.

6.3.2 Parameteranalyse

Entscheidend für die Qualität der Klassifizierung mit einem Random Forest sind die Anzahl k der Entscheidungsbäume sowie die Anzahl m an zufällig ausgewählten Merkmalen (vgl. Abschnitt 2.5.3). Dazu wurden mehrere Random Forests mit unterschiedlicher Parametrisierung auf der Trainingsdatenmenge $\mathcal{D}$ trainiert. Die resultierenden Out-of-Bag-Fehler (OOB-Fehler) sind in Abb. 6.6(a) dargestellt.

Wie in (Breiman 2001) festgestellt wurde ist ein Random Forest nicht besonders empfindlich bezüglich der Wahl von m. Mit Ausnahme von $m = 1$ liegen die OOB-Fehler im Bereich $[0,0030; 0,0035]$ was bei $|\mathcal{D}| = 39876$ Paaren einer Differenz von 20 falsch klassifizierten Paaren entspricht. Der Parameterwert $m = 5$ kann als optimal für $\mathcal{D}$ betrachtet werden, da er zu dem niedrigsten Klassifikationsfehler über alle k führt.

Ein Random Forest ist robust gegenüber einer Überanpassung bezüglich der Anzahl k an Entscheidungsbäumen. Damit ist k lediglich durch die zur Verfügung stehenden Ressourcen und die erforderliche Effizienz begrenzt (vgl. Abschnitt 2.5.3). Um jedoch eine zuverlässige Statistik zu



(a) Trainingsdaten $\mathcal{D}$

(b) Trainingsdaten $\mathcal{D}_1$

(c) Trainingsdaten $\mathcal{D}_2$

(d) Trainingsdaten $\mathcal{D}_3$

Abbildung 6.6: Out-of-Bag-Fehler mehrerer Random Forests in Abhängigkeit von der Anzahl k an Entscheidungsbäumen, der Anzahl m an zufällig ausgewählten Merkmalen sowie der Trainingsdatenmenge.

erhalten, sollte $k \geq 1000$ bis hin zu $k = 5000$ verwendet werden (Breiman 2004). Für $m = 5$ ist in Abb. 6.6(a) ab etwa $k = 1400$ keine signifikante Änderung des OOB-Fehlers mehr festzustellen.

Die Auswirkungen unterschiedlicher Bildauflösungen auf die Klassifikation sind in den Abb. 6.6(b) bis 6.6(d) dargestellt. Es ist zu beobachten, dass sich die Klassifikationsgenauigkeit mit steigender Bildauflösung verbessert. Die Ursache hierfür liegt in der höheren Anzahl an Korrespondenzen, die aufgrund feinerer Bildauflösungen ermittelt werden konnten. Die höhere Anzahl ermöglicht letztendlich eine zuverlässigere Bestimmung von Statistiken für die Klassifikationsmerkmale.

6.3.3 Merkmalsanalyse

Die Bedeutung einzelner Merkmale hinsichtlich der Klassifizierung (vgl. Abschnitt 2.5.3) ist in Abb. 6.7 dargestellt. Die höchste Bedeutung haben demnach die Merkmale R und D, gefolgt von

V und K. Die restlichen Merkmale weisen hingegen eine deutlich geringere Bedeutung auf. Dies ist auch aus ihren einzelnen Klassifikationsfehlern ersichtlich (siehe Abschnitt 6.3.4), die zeigen, dass sie keine gute Aufteilung der Trainingsdaten ermöglichen. K weist, trotz guter Unterteilung der Trainingsdaten, eine relative geringe Bedeutung auf. Dies liegt, wie in (Michelini und Mayer 2014) gezeigt wurde, an der starken Korrelation mit R, wobei R etwas mehr Informationsgehalt bietet und dadurch vermutlich während der Trainingsphase des Random Forests bevorzugt ausgewählt wird.

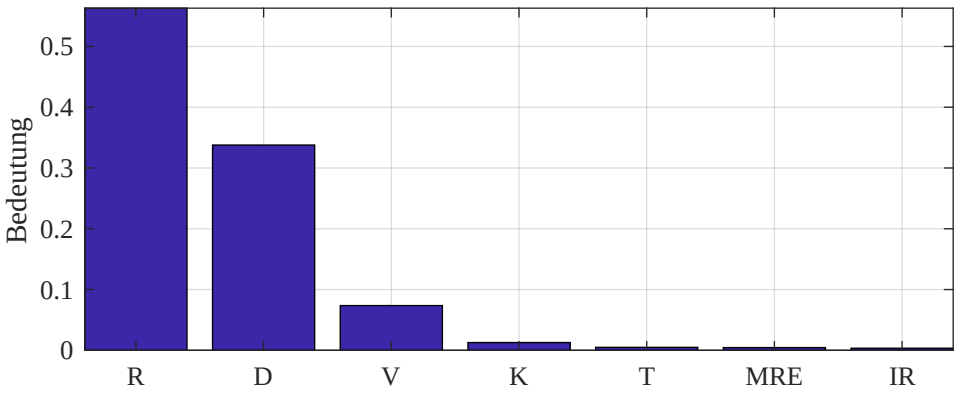


Abbildung 6.7: Bedeutung einzelner Merkmale im Random Forest mit $k = 3000$ und $m = 5$, trainiert an der Trainingsdatenmenge $\mathcal{D}$. Ein höherer Wert bedeutet eine höhere Bedeutung hinsichtlich der Klassifizierung.

Über Ausschluss der Merkmale mit geringer Bedeutung aus dem Trainingsprozess erfolgte eine weitere Analyse ihrer Relevanz für die Klassifizierung. Wie Abb. 6.8 zeigt, erhöht sich der Klassifikationsfehler mit jedem Ausschluss. Lediglich bei den Merkmalen T und MRE zeigte sich keine gravierende Veränderung der Klassifikationsgenauigkeit. Somit ist für eine möglichst genaue Klassifizierung die Benutzung aller Merkmale vorteilhaft.

6.3.4 Klassifizierung

Der Random Forest Klassifikator trainiert auf der Trainingsdatenmenge $\mathcal{D}$ wird mit den Parametern $m = 5$ und $k = 3000$ zur Detektion kritischer Kamerakonfigurationen eingesetzt. Die optimalen Parameter wurden im Abschnitt 6.3.2 ermittelt, wobei mit $k = 3000$ eine höhere Anzahl an Entscheidungsbäumen gewählt wurde, um zuverlässigere Klassifikationswahrscheinlichkeiten zu erhalten.

Der Klassifikator weist einen Klassifikationsfehler in Form des OOB-Fehlers von $0,31\%$ auf. Im Vergleich dazu führen die Klassifikationen anhand einzelner Merkmale unter Verwendung eines optimalen Schwellenwerts zu deutlich größeren Klassifikationsfehlern ($R \rightarrow 1,3\%$, $D \rightarrow 1,3\%$, $V \rightarrow 2,0\%$, $K \rightarrow 1,2\%$, $T \rightarrow 33,5\%$, $MRE \rightarrow 35,7\%$, $IR \rightarrow 34,0\%$). Somit ermöglicht die Kombination dieser Merkmale in einem Random Forest Klassifikator eine deutliche Verbesserung der Klassifikationsgenauigkeit.

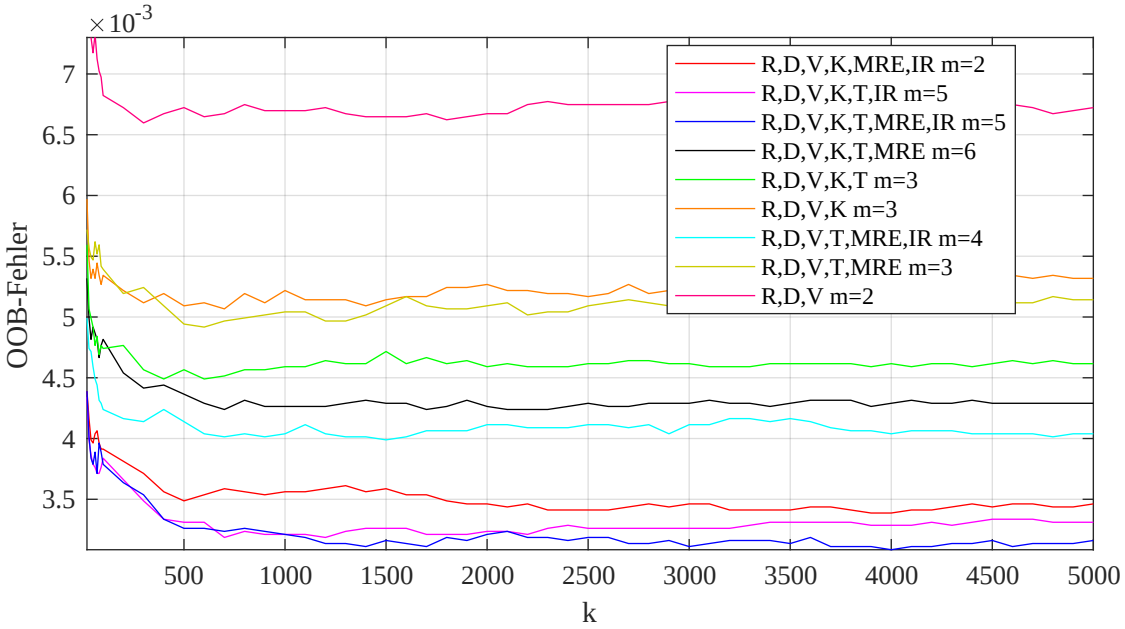


Abbildung 6.8: OOB-Fehler für unterschiedliche Merkmalskombinationen in Abhängigkeit von der Anzahl k an Entscheidungsbäumen. Die Bestimmung der optimalen Anzahl m an zufällig ausgewählten Merkmale erfolgte basierend auf dem niedrigsten Median des OOB-Fehlers.

Die Trainingsdaten bestehen ausschließlich aus degenerierten und nicht degenerierten Paaren. Kritische Paare sind in den Trainingsdaten nicht explizit enthalten, weil ihre Generierung äußerst schwierig ist. Außerdem gibt es eine Vielzahl an Einflüssen, die zu kritischen Paaren führen können, sodass eine allgemeine Berücksichtigung nicht möglich ist. Somit ist eine direkte Klassifizierung in drei Klassen mittels Random Forest nicht möglich. Stattdessen erfolgt die Klassifizierung in zwei Klassen, nämlich stabile und instabile Paare (vgl. Abschnitt 4.2). Basierend darauf und der Wahrscheinlichkeit der Klassenzugehörigkeit wird eine weitere Klassifizierung in kritische Paare vorgenommen. Ein Paar wird dabei als kritisch eingestuft, wenn seine Klassifikation nicht zuverlässig erfolgen kann.

Eine Möglichkeit kritische Paare zu ermitteln wäre die Anwendung eines Schwellenwerts für die Klassenwahrscheinlichkeit, bei der die zugehörige Klassifikation als zuverlässig einzustufen ist. Allerdings ergaben praktische Untersuchungen, dass auch Paare vorkommen können, die mit hoher Wahrscheinlichkeit falsch klassifiziert wurden. In diesem Fall hilft die Anwendung eines festen Schwellenwerts nur bedingt.

Stattdessen werden kritische Paare einer erneuten Verifikation unter Verwendung einer höheren Bildauflösung unterzogen (siehe Abschnitt 5.2). Wie in Abschnitt 6.3.2 festgestellt wird hierdurch tendenziell eine genauere Klassifikation erzielt. Die aufwändige Verifikation ist zudem nur an einer kleinen Teilmenge an kritischen Paaren zu wiederholen. Wenn sich die Klassenwahrscheinlichkeit nach erneuter Verifikation erhöht, wird das Paar der entsprechenden Klasse zugeordnet, ansonsten als kritisch belassen. Diese Vorgehensweise führte zu einer besseren Detektion von kritischen Kamerakonfigurationen unter Reduzierung von Fehlklassifikationen.

Bildmenge	Geschlossene Bildschleifen	Dämpfungsfaktor	Punkte	σ_0
UQ St Lucia	keine	–	203466	0,228
	1	10	204485	0,236
	55	0	199373	0,253
Dorf-Überflug	keine	–	166993	0,283
	4	10	170472	0,342
	110	0	199182	0,422

Tabelle 6.4: Auswirkungen vom Schleifenschluss auf die Bildmengen *UQ St Lucia* und *Dorf-Überflug* bezüglich der Anzahl der rekonstruierten 3D-Punkte sowie des mittleren Rückprojektionsfehlers σ_0 in Pixel.

6.4 Schleifenschluss

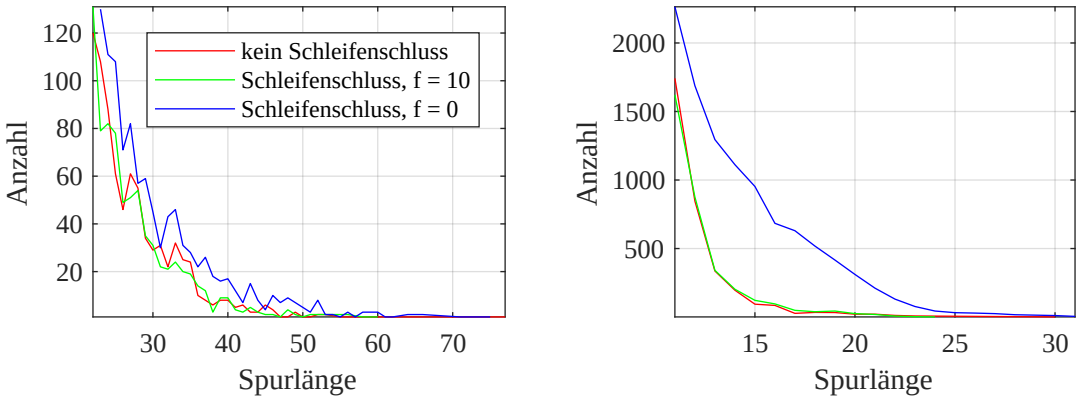
Das Schließen von Bildschleifen ist insbesondere bei langen, zyklischen Bildsequenzen essenziell für eine genaue Bestimmung von Kameraposen. Um diese Fähigkeit zu demonstrieren, erfolgt in diesem Abschnitt die Untersuchung des entwickelten Ansatzes zum Schließen von Bildschleifen (Abschnitt 4.8) anhand der Bildmengen *UQ St Lucia* und *Dorf-Überflug*. Diese enthalten einfache Bildschleifen und sind gut geeignet, um die Auswirkungen darzustellen.

Aus Effizienzgründen wird während der hierarchischen Vereinigung (Mayer 2014) eine Punktreduktion durchgeführt, bei der zufällig 3D-Punkte aus der Bündelausgleichung ausgeschlossen werden. Für die Evaluierung wurde diese Punktreduktion deaktiviert, um die Anzahl an rekonstruierten 3D-Punkten direkt vergleichen zu können.

Änderungen bezüglich der Anzahl der 3D-Punkte sowie des Rückprojektionsfehlers aufgrund von geschlossenen Bildschleifen sind in Tabelle 6.4 aufgelistet. Der Einfluss des Schleifenschlusses auf den Rückprojektionsfehler ist komplex. Ein wichtiger Grund für den schlechteren Wert ist die Verwendung der nichtlinearen Bündelausgleichung, die den Fehler nur lokal verringern kann. Des Weiteren sind die höheren Rückprojektionsfehler realistischer durch im Durchschnitt längere Spurlängen (siehe unten).

Die geringere Anzahl der 3D-Punkte nach dem Schleifenschluss resultiert daraus, dass zuvor oft der gleiche 3D-Punkt aufgrund von fehlenden Verknüpfungen zwischen den Bildern mehrfach rekonstruiert wurde. Nach dem Schleifenschluss entstand daraus dann jeweils ein 3D-Punkt. Eine weitere Ursache liegt auch in der genaueren internen Überprüfung dank Mehrfachverknüpfungen, wodurch fehlerhafte 3D-Punkte besser detektiert und herausgefiltert werden können.

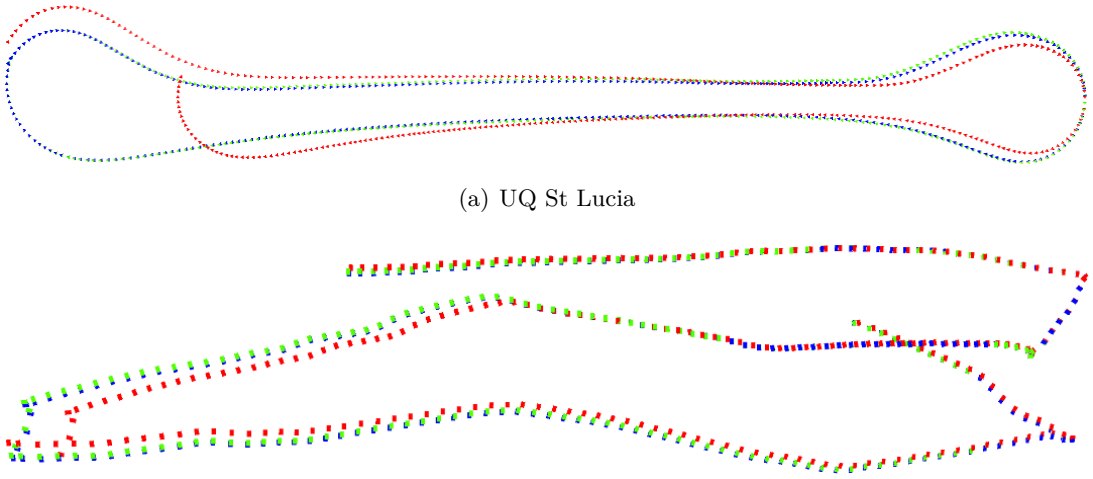
Der Einfluss auf die Spurlängen ist in Abb. 6.9 dargestellt. Eine *Spur* (engl. *track*) entspricht der Verknüpfung korrespondierender Punkte mit dem selben 3D-Punkt über mehrere Bilder. Die Anzahl der Bilder, in denen ein 3D-Punkt sichtbar ist, bezeichnet man als *Spurlänge*. Erwartungsgemäß steigt die Anzahl an längeren Spuren mit der Anzahl an Schleifenschlüssen. Aufgrund von mehrdeutigen 3D-Punkten vor dem Schleifenschluss, kommt es nach diesem zu einer leichten Verringerung der maximalen Spurlänge bei der Bildmenge *UQ St Lucia*.



(a) UQ St Lucia (b) Dorf-Überflug

Abbildung 6.9: Vergleich der Spurlängen für die Bildmengen *UQ St Lucia* und *Dorf-Überflug* mit und ohne Schleifenschluss.

In Abb. 6.10 sind die Abweichungen in den geschätzten Kameraposen mit und ohne Schleifenschluss zu sehen. Ohne geschlossene Bildschleifen kommt es bei der Bildmenge *UQ St Lucia* zu einer deutlichen Abweichung an den Schleifenenden, die zu falschen Kameraposen führen. Bei der Bildmenge *Dorf-Überflug* sind die Abweichungen etwas geringer, führen jedoch zusammen mit der mehrfachen Rekonstruktion derselben 3D-Punkte zu doppelten 3D-Strukturen in der Punktwolke. Diese zeigt Abb. 6.11 in Form von doppelten Dächern und Boden.



(a) UQ St Lucia

(b) Dorf-Überflug

Abbildung 6.10: Kameraposen für die Bildmengen *UQ St Lucia* und *Dorf-Überflug* dargestellt in Rot ohne Schleifenschluss, nach dem Schleifenschluss mit Dämpfungsfaktor $f = 10$ in Grün und mit $f = 0$ in Blau.

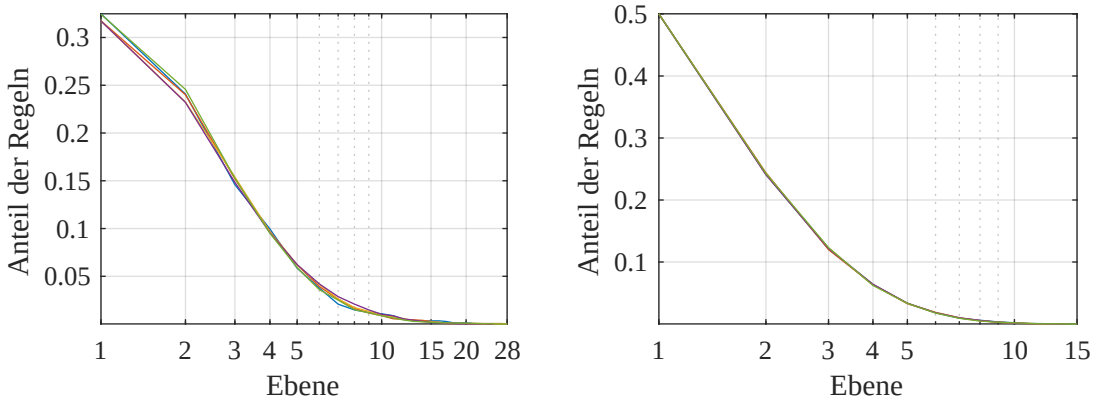
(a) Ohne Schleifenschluss

(b) Mit Schleifenschluss ($f = 10$)

Abbildung 6.11: Vergleich der rekonstruierten Punktwolken für die Bildmenge *Dorf-Überflug* mit und ohne Schleifenschluss.

6.5 Vereinigung von Bildteilmengen

Um eine bessere Lastverteilung auf modernen Multiprozessorsystemen zu ermöglichen, erfolgt die Generierung der Vereinigungsregeln über die Bestimmung der Paarung im Verknüpfungsgraphen (vgl. Abschnitt 5.7.2). Dieser Abschnitt vergleicht diesen Ansatz mit der clusterbasierten Vorgehensweise in (Mayer 2014) hinsichtlich des erzielbaren Parallelisierungsgrades.



(a) Clusterbasierte Generierung

(b) Paarungsbasierte Generierung

Abbildung 6.12: Anteil der Regeln pro Ebene in Regelbäumen für die Bildmengen *Kirche*, *Kloster*, *Übungsplatz*, *Flugplatz* und *Dorf* in Abhängigkeit von der eingesetzten Generierungsmethode. Verschiedene Linienfarben kennzeichnen die korrespondierenden Bildmengen.

In Abb. 6.12 ist die Verteilung der Regeln über die Ebenen der Regelbäume für einige größere Bildmengen dargestellt. Die paarungsbasierte Generierung führt zu deutlich mehr Regeln in niedrigeren Ebenen. Dies resultiert in flacheren Regelbäumen, die, aufgrund von mehr Regeln pro Ebene einen höheren Parallelisierungsgrad ermöglichen. Die niedrigere Höhe des Regelbaumes bietet zudem eine bessere Möglichkeit, über Ebenen hinweg unabhängige Regeln parallel zu verarbeiten.

6.6 Vergleich mit existierenden Verfahren

In diesem Abschnitt erfolgt der Vergleich des im Rahmen dieser Arbeit entwickelten Ansatzes, genannt *Automatic Pose Estimator (APE)* mit folgenden existierenden Verfahren:

VisualSFM[1] (Wu 2013) ist ein inkrementelles Verfahren, das die Kameraposeschätzung mittels Parallelisierung auf der Grafikkarte beschleunigt. Aufgrund der Untersuchung aller möglichen Paarkombinationen weist es eine hohe Komplexität auf, weswegen es für größere Bildmengen unzureichend skaliert. In dem Vergleich dient dieses Verfahren zum Aufzeigen von Grenzen der ausschließlichen Nutzung effizienter Implementierung.

COLMAP[2] ist ein von Schönberger und Frahm (2016) entwickeltes inkrementelles Verfahren, das Techniken von *VisualSFM* (Wu 2013) sowie einen Vocabulary Tree (Schönberger et al. 2016) zur Beschleunigung der Kameraposeschätzung einsetzt. Für den Vergleich werden bereitgestellte Vocabulary Trees entsprechend der Bildmengengröße benutzt. Aufgrund der effizienten Umsetzung und der Anwendung aktueller Techniken wird es in der Evaluierung als das Referenzverfahren betrachtet.

OpenMVG[3] beinhaltet das inkrementelle Verfahren von Moulon et al. (2012), das für den Umgang mit größeren Bildmengen Cascade Hashing (Cheng et al. 2014) verwendet. Für den Vergleich zeigt es ein Verfahren, das mit Cascade Hashing eine alternative Technik zum Vocabulary Tree und der in dieser Arbeit entwickelten Deskriptoreinbettung zur Effizienzsteigerung einsetzt.

SAMANTHA stellt ein hierarchisches Verfahren basierend auf dem Ansatz von Toldo et al. (2015) dar. Es gehört zur kommerzieller Photogrammetriesoftware 3DF Zephyr[4] und ermöglicht einen Vergleich mit einem zweiten hierarchischen Verfahren.

Bis auf *SAMANTHA* basieren alle Verfahren auf SIFT-Merkmalen (Lowe 2004). Aus lizenzrechtlichen Gründen benutzt *SAMANTHA* Merkmale, die entsprechend (Lindeberg 1998) ähnlich zu SIFT extrahiert wurden. Weitere Eigenschaften der Verfahren sind in Tabelle 6.5 gegenüber gestellt.

Für den Vergleich wird keine Zusatzinformation in Form von GPS-Daten oder Aufnahmekonfiguration benutzt. Die Kalibrierung wird automatisch anhand der Exif-Tags in den Bildern bestimmt. Außerdem werden bei allen Verfahren Standardeinstellungen verwendet. Hierbei ist zu beachten, dass die Ergebnisse des Vergleichs in Tabelle 6.6 durch die Anpassung der Einstellungen einzelner Verfahren auch anders ausfallen können. Dies spielt jedoch eine untergeordnete Rolle, da es hierbei weniger um einen direkten Vergleich der einzelnen Verfahren, sondern vielmehr um die Einordnung von *APE* in den aktuellen Stand der Forschung geht.

Die besten Ergebnisse konnten mit *APE* und *COLMAP* erzielt werden. Während *APE* bessere Laufzeiten aufweist, ist *COLMAP*, mit Ausnahme der Bildmenge *UQ St Lucia*, meist in der Lage,

[1] `ccwu.me/vsfm`, Version 0.5.26
[2] `github.com/colmap/colmap`, Version 3.2
[3] `github.com/openMVG/openMVG`, Version 1.2.0
[4] `3dflow.net`, 3DF Zephyr Aerial, Version 3.503

Verfahren	Merkmals-extraktion	Bildzuordnung	Bündel-ausgleichung	Vereinigung
APE	GPU (Wu 2012)	Deskriptoreinbettung CPU	CPU	hierarchisch
COLMAP	GPU (Wu 2012)	Vocabulary Tree GPU (Wu 2012)	GPU (Wu et al. 2011)	inkrementell
VisualSFM	GPU (Wu 2012)	GPU (Wu 2012)	GPU (Wu et al. 2011)	inkrementell
OpenMVG	CPU	Cascade Hashing (Cheng et al. 2014) CPU	CPU	inkrementell
SAMANTHA	GPU	GPU	CPU	hierarchisch

Tabelle 6.5: Eigenschaften der Verfahren

etwas größere Blöcke zu bilden. Diese beiden Eigenschaften sind jedoch voneinander abhängig. Das bedeutet, die Laufzeit steigt mit dem Aufwand für die Suche nach fehlenden Verknüpfungen.

Interessanterweise hatten alle Verfahren bis auf *APE* und *SAMANTHA* Probleme mit der Bildmenge *UQ St Lucia*. Trotz vorgegebener Kalibrierung waren sie nicht in der Lage, alle Bilder zu verknüpfen. Bei *VisualSFM* scheiterte die Kameraposeschätzung sogar komplett, was am sehr großen Rückprojektionsfehler zu erkennen ist.

Insgesamt zeigte *VisualSFM* die schlechtesten Eigenschaften. Neben der Bildmenge *UQ St Lucia* konnten auch *Flugplatz* und *Dorf* nicht erfolgreich verarbeitet werden, weil es zum undefinierten Abbruch während der Ausführung kam. Zudem wurden nur relativ kleine Blöcke gebildet. Während der Kameraposeschätzung für die Bildmengen *Kirche* und *Übungsplatz* kam es zu Fehlermeldungen über unzureichende Punktanzahl.

OpenMVG skaliert unzureichend für größere Bildmengen. Einer der Gründe hierfür könnte in der großen Anzahl an Merkmalen liegen, die zwar zu deutlich mehr rekonstruierten 3D-Punkten im Falle der Bildmengen *Kirche* und *Kloster* führen, aber auch den Rechenaufwand erhöhen. Zudem führt die fehlende Unterstützung für Grafikkarten dazu, dass selbst die Merkmalsextraktion eine vergleichsweise enorme Rechenzeit erfordert.

Der clusterbasierte, hierarchische Ansatz von *SAMANTHA* wurde aus bisher ungeklärten Gründen ab etwa drei Viertel der Verarbeitung sehr langsam. Somit konnten die Kameraposen der Bildmenge *Übungsplatz* aber auch *Flugplatz* und *Dorf* selbst nach einer Woche nicht geschätzt werden.

6.7 Bewertung der erzielten Ergebnisse

Die Experimente mit dem entwickelten Verfahren für die automatische Kameraposeschätzung erfolgen mit kleineren sowie größeren komplexen Bildmengen aus Abschnitt 6.1. Die Vergleiche

mit bereits existierenden Verfahren im Abschnitt 6.6 zeigen, dass das entwickelte Verfahren hinsichtlich seiner Fähigkeiten und Effizienz dem Stand der Forschung entspricht.

Die Einbettung von Deskriptoren (Abschnitt 4.1.1) ermöglicht eine effiziente Schätzung von Bildähnlichkeiten ohne signifikante Qualitätseinbußen (vgl. Abschnitt 6.2). Somit konnte sie an Stelle von Vocabulary Trees oder ähnlichen Techniken erfolgreich eingesetzt werden, um größere Bildmengen zu verarbeiten. Allerdings lässt sich hiermit die Komplexität nur um einen konstanten Faktor reduzieren, sodass mit steigender Größe der Bildmengen andere Techniken effizienter sein könnten.

Über die Klassifizierung mittels Random Forest konnten Paare mit Bewegungsdegeneration zuverlässig ermittelt werden (vgl. Abschnitt 6.3). Allerdings ist eine strikte Klassifizierung in Paare mit und ohne Bewegungsdegeneration oft nicht ausreichend. Vielmehr sind auch Paare mit unzureichenden Basen, die in komplexen Bildmenge weit häufiger vorkommen, herauszufiltern. Da für solche Konfigurationen Trainingsdaten schwer zu generieren sind, erfolgte ihre Detektion über eine zweifache Klassifikation auf unterschiedlichen Bildauflösungen. Dies erhöht zwar geringfügig den Rechenaufwand, führt jedoch zur zuverlässigeren Klassifikation von kritischen Kamerakonfigurationen.

Die Verwendung einer minimalen Teilmenge an Triplets für die Konstruktion eines Blocks (Abschnitt 4.7) führt dazu, dass Bildschleifen explizit geschlossen werden müssen. Die Evaluierung im Abschnitt 6.4 zeigt, dass der entwickelte Ansatz aus Abschnitt 4.8 dazu geeignet ist. Der Dämpfungsfaktor bestimmt die Komplexität und indirekt auch die Anzahl an zu schließenden Bildschleifen. Wie in Abschnitt 2.1.7 beschrieben und durch die Experimente bestätigt, ist es oft ausreichend, nur größere Bildschleifen zu schließen. Dadurch lässt sich die Komplexität über einen größeren Dämpfungsfaktor reduzieren ohne die Qualität der geschätzten Kameraposen negativ zu beeinflussen.

Schließlich wurde in Abschnitt 6.5 gezeigt, dass die Generierung von Vereinigungsregeln über die Paarung im Verknüpfungsgraphen (Abschnitt 5.7) geeigneter ist als die in (Mayer 2014) eingesetzte clusterbasierte Vorgehensweise. Der resultierende Regelbaum wies eine deutlich höhere Anzahl an Vereinigungsregeln in niedrigeren Ebenen auf, was zu einer geringeren Höhe führte. Dies ermöglicht einen höheren Parallelisierungsgrad und bietet aufgrund feinerer Granularität auch eine höhere Unabhängigkeit zwischen Vereinigungsoperationen unterschiedlicher Ebenen. Letzteres führt nochmals zur Steigerung des Parallelisierungsgrads.

Bildmenge	Verfahren	t_m	t_v	t_r	t	$\frac{B}{N}$	Punkte	σ_0
Dorf-Überflug (232 Bilder)	APE	0,01	0,05	0,02	0,08	1,000	46150	0,29
	COLMAP	0,02	0,63	0,41	1,06	1,000	229431	0,69
	VisualSFM	0,03	0,24	0,02	0,29	0,600	22570	1,04
	OpenMVG	0,21	0,19	0,74	1,14	1,000	442261	0,39
	SAMANTHA	0,09	0,04	0,17	0,30	1,000	211857	1,22
Piazza Bra (331 Bilder)	APE	0,01	0,35	0,03	0,39	0,940	59845	0,36
	COLMAP	0,02	0,54	0,60	1,16	0,997	185779	0,61
	VisualSFM	0,02	1,15	0,03	1,20	0,284	21462	0,98
	OpenMVG	0,17	0,13	0,32	0,62	0,961	173161	0,37
	SAMANTHA	0,08	0,09	0,13	0,30	0,931	169685	1,55
UQ St Lucia (351 Bilder)	APE	0,01	0,18	0,03	0,22	1,000	61289	0,23
	COLMAP	0,01	0,38	0,72	1,11	0,530	32206	0,51
	VisualSFM	0,02	1,29	0,33	1,64	0,430	14295	205,62
	OpenMVG	0,03	0,07	0,02	0,12	0,460	7615	0,58
	SAMANTHA	0,01	0,02	0,17	0,20	1,000	36639	0,31
Kirche (1455 Bilder)	APE	0,12	0,38	0,11	0,61	0,999	290748	0,55
	COLMAP	0,16	2,56	1,59	4,31	0,999	560887	0,95
	VisualSFM	0,21	10,06	0,09	10,35	0,198	37476	0,74
	OpenMVG	2,41	9,21	9,85	21,47	0,994	1684501	0,52
	SAMANTHA	0,83	0,32	4,33	5,47	0,970	156329	3,31
Kloster (1769 Bilder)	APE	0,48	1,04	0,14	1,66	0,992	330749	0,63
	COLMAP	0,24	2,27	2,75	5,26	0,997	756306	0,98
	VisualSFM	0,29	13,38	0,14	13,81	0,201	20585	1,36
	OpenMVG	3,68	13,03	6,88	23,59	0,503	1415697	0,58
	SAMANTHA	1,10	1,28	9,97	12,35	0,994	789725	8,24
Übungsplatz (3664 Bilder)	APE	3,42	2,14	1,32	6,88	0,997	701765	0,59
	COLMAP	0,69	6,30	7,96	14,94	0,999	3038212	1,17
	VisualSFM	0,80	76,82	0,32	77,93	0,165	49845	1,50
	OpenMVG	12,47	>144	–	–	–	–	–
	SAMANTHA	3,70	5,19	>144	–	–	–	–
Flugplatz (6210 Bilder)	APE	4,18	6,91	2,37	13,46	0,985	1191525	0,50
	COLMAP	0,90	10,53	35,05	46,48	0,999	6099781	1,17
	VisualSFM	1,12	209,82	×	–	–	–	–
Dorf (6405 Bilder)	APE	1,43	4,34	1,92	7,69	0,976	1318265	0,38
	COLMAP	0,50	13,38	13,12	26,99	0,992	3741066	0,85
	VisualSFM	0,62	222,25	×	–	–	–	–

Tabelle 6.6: Ergebnisse der automatischen Kameraposeschätzung für die Bildmengen aus Abschnitt 6.1. Die Laufzeiten für die Merkmalsextraktion t_m, die Verknüpfungsbestimmung t_v, die Vereinigung t_r sowie die Gesamtlaufzeit $t = t_m + t_v + t_r$ sind in Stunden angegeben. B bezeichnet die Anzahl der Bilder im größten Block und N die Größe der jeweiligen Bildmenge. Die letzten beiden Spalten geben die Anzahl der rekonstruierten 3D-Punkte und den mittleren Rückprojektionsfehler σ_0 in Pixel für den größten Block an.

7 Fazit und Ausblick

Im Rahmen dieser Dissertation wurde ein Verfahren für die automatische Kameraposeschätzung in komplexen Bildmengen entwickelt. Dieses benötigt, außer der Kamerakalibrierung, keine Zusatzinformation wie beispielsweise Aufnahmekonfiguration, GPS- oder INS-Daten. Da Kameraparameter meist automatisch aus den Metadaten der Bilder ermittelbar sind und die Bildverknüpfungen automatisch hergestellt werden, erfordert die Kameraposeschätzung keine manuellen Eingriffe.

Basierend auf den hergeleiteten theoretischen Konzepten erfolgte die praktische Umsetzung des iterativen, graphenbasierten Ansatzes. Seine Fähigkeit effizient mit komplexen Aufnahmekonfigurationen in größeren Bildmengen umgehen zu können wurde anhand von Experimenten belegt. Letztere führten außerdem über den Vergleich mit einigen aktuellen Verfahren für die Kameraposeschätzung zur Einordnung in den Stand der Forschung.

Eine interessante Fragestellung für die zukünftige Forschung ist, ob sich durch die Ausnutzung spezieller Eigenschaften des Bildteilmengengraphen geeignetere Vereinigungsregeln hinsichtlich der Blockstabilität generieren lassen. In der vorliegenden Arbeit werden Schleifenschlüsse durch Kreise im Bildteilmengengraphen repräsentiert (vgl. Abschnitt 5.7). Abhängig vom Dämpfungsfaktor können dabei mehrere größere Kreise aber auch kleinere vorkommen, die unter Umständen Teilkreise größerer Kreise sind. Für eine stabilere Vereinigung wäre es wahrscheinlich vorteilhaft, erstmal die Bilder einzelner Kreise zu vereinigen und hierbei zuerst mit kleineren Kreisen bzw. solchen mit stärkeren Verknüpfungen zwischen den Bildern anzufangen. Die Kreise könnten im Bildteilgraphen detektiert und basierend auf den Spurlängen gewichtet werden, um bestimmte Vereinigungen zu priorisieren. Aus Effizienzgründen würde es aller Voraussicht nach ausreichen, sich hierzu auf die fundamentale Menge der Kreise (Kavitha et al. 2009) des Bildteilmengengraphen zu beschränken.

Die Effizienz aber auch die Robustheit des entwickelten Verfahrens lassen sich durch die Verwendung von Zusatzinformation erhöhen. Zukünftige Arbeiten werden sich daher mit der Einschränkung des Suchraumes während der Bestimmung der Bildverknüpfung durch Ausnutzung von GPS-Daten sowie Aufnahmekonfigurationen beschäftigen. In diesem Zusammenhang ist auch zu untersuchen, inwieweit sich die Verfahren von Roth et al. (2017) oder MODS (Mishkin et al. 2015) zur Steigerung der Robustheit gegenüber extremen Bildverzerrungen effizient einsetzen lassen.

Literaturverzeichnis

Agapito, L., E. Hayman und I. D. Reid (2001). "Self-Calibration of Rotating and Zooming Cameras". In: *International Journal of Computer Vision* 45.2, S. 107–127.

Agarwal, S., N. Snavely, S. M. Seitz und R. Szeliski (2010). "Bundle Adjustment in the Large". In: *European Conference on Computer Vision*.

Agarwal, S., N. Snavely, I. Simon, S. M. Seitz und R. Szeliski (2009). "Building Rome in a Day". In: *International Conference on Computer Vision*.

Bay, H., A. Ess, T. Tuytelaars und L. V. Gool (2006). "SURF: Speeded-Up Robust Features". In: *European Conference on Computer Vision*.

Beder, C. und R. Steffen (2006). "Determining an Initial Image Pair for Fixing the Scale of a 3D Reconstruction from an Image Sequence". In: *Deutsche Arbeitsgemeinschaft für Mustererkennung*.

Beis, J. S. und D. G. Lowe (1997). "Shape Indexing using Approximate Nearest-Neighbour Search in High-Dimensional Spaces". In: *Conference on Computer Vision and Pattern Recognition*.

Bellman, R. (1957). *Dynamic Programming*. Princeton University Press.

Biau, G. und E. Scornet (2016). "A Random Forest Guided Tour". In: *TEST* 25 (2), S. 197–227.

Breiman, L. (1996). "Bagging Predictors". In: *Machine Learning* 24 (2), S. 123–140.

– (2001). "Random Forests". In: *Machine Learning* 45 (1), S. 5–32.

– (2004). *Manual on Setting Up, Using and Understanding Random Forests V4.0*. URL: `https://www.stat.berkeley.edu/~breiman/Using_random_forests_v4.0.pdf`.

Breiman, L., J. Friedman, C. J. Stone und R. A. Olshen (1984). *Classifikation and Regression Trees*. Chapman und Hall/CRC.

Charikar, M. S. (2002). "Similarity Estimation Techniques from Rounding Algorithms". In: *ACM Symposium on Theory of Computing*.

Chen, Y. H. (2011). "An Improved Approximation Algorithm for the Terminal Steiner Tree Problem". In: *Computational Science and Its Applications – ICCSA*. Band 6784, S. 141–151.

Cheng, J., C. Leng, J. Wu, H. Cui und H. Lu (2014). "Fast and Accurate Image Matching with Cascade Hashing for 3D Reconstruction". In: *Conference on Computer Vision and Pattern Recognition*.

Chum, O., J. Matas und J. Kittler (2003). "Locally Optimized RANSAC". In: *Deutsche Arbeitsgemeinschaft für Mustererkennung*.

Chum, O., A. Mikulík, M. Perdoch und J. Matas (2011). "Total Recall II: Query Expansion Revisited". In: *Conference on Computer Vision and Pattern Recognition*.

Chum, O., J. Philbin, J. Sivic, M. Isard und A. Zisserman (2007). "Total Recall: Automatic Query Expansion with a Generative Feature Model for Object Retrieval". In: *International Conference on Computer Vision*.

Chum, O., T. Werner und J. Matas (2005). "Two-View Geometry Estimation Unaffected by a Dominant Plane". In: *Conference on Computer Vision and Pattern Recognition*.

Cormen, T. H., C. E. Leiserson, R. Rivest und C. Stein (2013). *Algorithmen – Eine Einführung*. Band 4. Oldenbourg Verlag München.

Crandall, D., A. Owens, N. Snavely und D. Huttenlocher (2011). "Discrete-Continuous Optimization for Large-Scale Structure from Motion". In: *Conference on Computer Vision and Pattern Recognition*.

Criminisi, A. und J. Shotton (2013). *Decision Forests for Computer Vision and Medical Image Analysis*. Springer London.

David, S. (1974). "Graphs with 1-Factors". In: *American Mathematical Society*.

Dempster, A. P., N. M. Laird und D. B. Rubin (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm". In: *Journal of the Royal Statistical Society* 39.1, S. 1–38.

Díaz-Uriarte, R. und S. A. de Andrés (2006). "Gene Selection and Classification of Microarray Data using Random Forest". In: *BMC Bioinformatics* 7.1.

Drake, D. E. und S. Hougardy (2004). "On Approximation Algorithms for the Terminal Steiner Tree Problem". In: *Information Processing Letters* 89 (1), S. 15–18.

Dreyfus, S. E. und R. A. Wagner (1971). "The Steiner Problem in Graphs". In: *Networks* 1.3, S. 195–207.

Edmonds, J. (1965). "Paths, Trees and Flowers". In: *Canadian Journal of Mathematics* 17, S. 449–467.

Enqvist, O., F. Kahl und C. Olsson (2011). "Non-Sequential Structure from Motion". In: *International Conference on Computer Vision Workshop*.

Farenzena, M., A. Fusiello und R. Gherardi (2009). "Structure-and-Motion Pipeline on a Hierarchical Cluster Tree". In: *International Conference on Computer Vision Workshop*.

Farenzena, M., A. Fusiello, R. Gherardi und R. Told (2008). "Towards Unsupervised Reconstruction of Architectural Models". In: *Vision, Modeling and Visualization Conference*.

Faugeras, O., Q. T. Lount und T. Papadopoulo (2004). *The Geometry of Multiple Images*. MIT Press.

Fischler, M. A. und R. C. Bolles (1981). "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". In: *Communications of the ACM* 24.6, S. 381–395.

Fitzgibbon, A. W. und A. Zisserman (1998). "Automatic Camera Recovery for Closed or Open Image Sequences". In: *European Conference on Computer Vision*.

Forssén, P.-E. (2007). "Maximally Stable Colour Regions for Recognition and Matching". In: *Conference on Computer Vision and Pattern Recognition*.

Förstner, W. (1984). "Quality Assessment of Object Location and Point Transfer Using Digital Image Correlation Techniques". In: 25.3A, S. 197–219.

Förstner, W. und E. Gülch (1987). "A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features". In: *ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data*.

Förstner, W. und B. P. Wrobel (2016). *Photogrammetric Computer Vision*. Springer.

Frahm, J.-M. und M. Pollefeys (2006). "RANSAC for (Quasi-)Degenerate Data (QDEGSAC)". In: *Conference on Computer Vision and Pattern Recognition*.

Frahm, J.-M. et al. (2010). "Building Rome on a Cloudless Day". In: *European Conference on Computer Vision*.

Friedman, J. H., J. L. Bentley und R. A. Finkel (1977). "An Algorithm for Finding Best Matches in Logarithmic Expected Time". In: *ACM Transactions on Mathematical Software* 3.3, S. 209–226.

Fuchs, B. (2003). "A Note on the Terminal Steiner Tree Problem". In: *Information Processing Letters* 87.4, S. 219–220.

Fuchs, B., W. Kern, D. Molle, S. Richter, P. Rossmanith und X. Wang (2007). "Dynamic Programming for Minimum Steiner Trees". In: *Theory of Computing Systems* 41.3, S. 493–500.

Fukunage, K. und P. M. Narendra (1975). "A Branch and Bound Algorithm for Computing k-Nearest Neighbors". In: *IEEE Transactions on Computers* 24.7, S. 750–753.

Fusiello, A., U. Castellani, L. Ronchetti und V. Murino (2002). "Model Acquisition by Registration of Multiple Acoustic Range Views". In: *European Conference on Computer Vision*.

Gherardi, R., M. Farenzena und A. Fusiello (2010). "Improving the Efficiency of Hierarchical Structure-and-Motion". In: *Conference on Computer Vision and Pattern Recognition*.

Gibson, S., J. Cook, T. Howard, R. Hubbold und D. Oram (2002). "Accurate Camera Calibratio for Off-line, Video-Based Augmented Reality". In: *International Symposium on Mixed and Augmented Reality*.

Gionis, A., P. Indyk und R. Motwani (1999). "Similarity Search in High Dimensions via Hashing". In: *International Conference on Very Large Data Bases*.

Grubbs, F. E. (1969). "Procedures for Detecting Outlying Observations in Samples". In: *Technometrics* 11.1, S. 1–21.

Grün, A. (1985). "Adaptive Least Squares Correlation: A Powerful Image Matching Technique". In: *South African Journal of Photogrammetry, Remote Sensing and Cartography* 14.3, S. 175–187.

Grünbaum, B. (2003). *Convex Polytopes*. 2. Auflage. Springer.

Hakimi, S. L. (1971). "Steiner's Problem in Graphs and its Implications". In: *Networks* 1.2, S. 113–133.

Hampel, F. R. (1971). "A General Qualitative Definition of Robustness". In: *The Annals of Mathematical Statistics* 42.6, S. 1887–1896.

Hampel, F. R., E. M. Ronchetti, P. J. Rousseeuw und W. A. Stahel (2005). *Robust Statistics: The Approach Based on Influence Functions*. Wiley-Interscience.

Harary, R. (1969). *Graph Theory*. Addison-Wesley.

Harris, C. und M. Stephens (1988). "A Combined Corner and Edge Detector". In: *Alvey Vision Conference*.

Hartley, R. (1997). "Lines and Points in Three Views and the Trifocal Tensor". In: *International Journal of Computer Vision* 22.2, S. 125–140.

Hartley, R., E. Hayman, L. de Agapito und I. Reid (1999). "Camera Calibration and the Search for Infinity". In: *International Conference on Computer Vision*.

Hartley, R. und A. Zisserman (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press.

Hartmann, W., M. Havlena und K. Schindler (2014). "Predicting Matchability". In: *Conference on Computer Vision and Pattern Recognition*.

– (2016). “Recent Developments in Large-scale Tie-point Matching”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 115, S. 47–62.

Hastie, T., R. Tibshirani und J. Friedman (2009). *The Elements of Statistical Learning*. Band 2. Springer.

Havlena, M., W. Hartmann und K. Schindler (2013). “Optimal Reduction of Large Image Databases for Location Recognition”. In: *International Conference on Computer Vision Workshop*.

Havlena, M. und K. Schindler (2014). “VocMatch: Efficient Multiview Correspondence for Structure from Motion”. In: *European Conference on Computer Vision*.

Havlena, M., A. Torii und T. Pajdla (2010). “Efficient Structure from Motion by Graph Optimization”. In: *European Conference on Computer Vision*.

Heath, K., N. Gelfand, M. Ovsjanikov, M. Aanjaneya und L. J. Guibas (2010). “Image Webs: Computing and Exploiting Connectivity in Image Collections”. In: *Conference on Computer Vision and Pattern Recognition*.

Heinly, J., E. Dunn und J.-M. Frahm (2012). “Comparative Evaluation of Binary Features”. In: *European Conference on Computer Vision*.

Heinly, J., J. L. Schönberger, E. Dunn und J.-M. Frahm (2015). “Reconstructing the World in Six Days”. In: *Conference on Computer Vision and Pattern Recognition*.

Hodge, V. und J. Austin (2004). “A Survey of Outlier Detection Methodologies”. In: *Artificial Intelligence Review* 22.2, S. 85–126.

Hougardy, S. und H. J. Prömel (1999). “A 1.598 Approximation Algorithm for the Steiner Problem in Graphs”. In: *ACM-SIAM Symposium on Discrete Algorithms*.

Huber, P. J. und E. M. Ronchetti (2009). *Robust Statistics*. John Wiley & Sons.

Indyk, P. und R. Motwani (1998). “Approximate Nearest Neighbors: Toward Removing the Curse of Dimensionality”. In: *ACM Symposium on Theory of Computing*.

Irani, M. und P. Anandan (1996). “Parallax Geometry of Pairs of Points for 3D Scene Analysis”. In: *European Conference on Computer Vision*.

Irschara, A., C. Hoppe, H. Bishof und S. Kluckner (2011). “Efficient Structure from Motion with Weak Position and Orientation Priors”. In: *Conference on Computer Vision and Pattern Recognition Workshops*.

Irschara, A., C. Zach, J.-M. Frahm und H. Bischof (2009). “From Structure-from-Motion Point Clouds to Fast Location Recognition”. In: *Conference on Computer Vision and Pattern Recognition*.

Jaccard, P. (1912). "The Distribution of the Flora in the Alpine Zone". In: *New Phytologist* 11.2, S. 37–50.

Jegou, H., M. Douze und C. Schmid (2008). "Hamming Embedding and Weak Geometric Consistency for Large Scale Image Search". In: *European Conference on Computer Vision*.

Jeong, Y., D. Nistér, D. Steedly, R. Szeliski und I.-S. Kweon (2012). "Pushing the Envelope of Modern Methods for Bundle Adjustment". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.8, S. 1605–1617.

Jian, Y.-D., D. C. Balcan und F. Dellaert (2011). "Generalized Subgraph Preconditioners for Large-Scale Bundle Adjustment". In: *International Conference on Computer Vision*.

Jiang, N., Z. Cui und P. Tan (2013). "A Global Linear Method for Camera Pose Registration". In: *International Conference on Computer Vision*.

Kähler, O. und J. Denzler (2006). "Detection of Planar Patches in Handheld Image Sequences". In: *Photogrammetric Computer Vision*.

Kanatani, K. (1996). *Statistical Optimization for Geometric Computation: Theory and Practice*. Elsevier.

Kangni, F. und R. Laganiére (2007). "Orientation and Pose Recovery from Spherical Panoramas". In: *International Conference on Computer Vision*.

Kavitha, T., C. Liebchen, K. Mehlhorn, D. Michail, R. Rizzi, T. Ueckerdt und K. A. Zweig (2009). "Cycle Bases in Graphs – Characterization, Algorithms, Complexity, and Applications". In: *Computer Science Review* 3.4, S. 199–243.

Ke, Y. und R. Sukthankar (2004). "PCA-SIFT: A More Distinctive Representation for Local Image Descriptors". In: *Conference on Computer Vision and Pattern Recognition*.

Kim, K. I., J. Tompkin, M. Theobald, J. Kautz und C. Theobalt (2012). "Match Graph Construction for Large Image Databases". In: *European Conference on Computer Vision*.

Klopschitz, M., A. Irschara, G. Reitmayr und D. Schmalstieg (2010). "Robust Incremental Structure from Motion". In: *International Symposium on 3D Data Processing, Visualization and Transmission*.

Koch, R., M. Pollefeys und L. V. Gool (1998). "Automatic 3D Model Acquisition from Uncalibrated Image Sequences". In: *Computer Graphics International Conference*.

Kou, L. T., G. Markowsky und L. C. Berman (1981). "A Fast Algorithm for Steiner Trees". In: *Acta Informatica* 15.2, S. 141–145.

Krig, S. (2014). "Interest Point Detector and Feature Descriptor Survey". In: *Computer Vision Metrics: Survey, Taxonomy, and Analysis*. ApressOpen. Kap. Interest Point Detector and Feature Descriptor Survey, S. 217–282.

Kruskal, J. B. (1956). "On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem". In: *Proceedings of the American Mathematical Society* 7.1, S. 48–50.

Leibe, B., K. Mikolajczyk und B. Schiele (2006). "Efficient Clustering and Matching for Object Class Recognition". In: *British Machine Vision Conference*.

Levi, N. und M. Werman (2003). "The Viewing Graph". In: *Conference on Computer Vision and Pattern Recognition*.

Li, X., C. Wu, C. Zach, S. Lazebnik und J.-M. Frahm (2008). "Modeling and Recognition of Landmark Image Collections Using Iconic Scene Graphs". In: *European Conference on Computer Vision*.

Liben-Nowell, D. und J. Kleinberg (2003). "The Link Prediction Problem for Social Networks". In: *International Conference on Information and Knowledge Management*.

Lin, G. und G. Xue (2002). "On the Terminal Steiner Problem". In: *Information Processing Letters* 84.2, S. 103–107.

Lindeberg, T. (1994). "Scale-space Theory: A Basic Tool for Analysing Structures at Different Scales". In: *Journal of Applied Statistics*. Band 21. 1–2, S. 225–270.

– (1998). "Feature Detection with Automatic Scale Selection". In: *International Journal of Computer Vision* 30.2, S. 79–116.

Longuet-Higgins, H. C. (1981). "A Computer Algorithm for Reconstructing a Scene from Two Projections". In: *Nature* 293, S. 133–135.

Lowe, D. G. (2004). "Distinctive Image Features from Scale-Invariant Keypoints". In: *International Journal of Computer Vision* 60.2, S. 91–110.

Manning, C. D., P. Raghavan und H. Schütze (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Martinec, D. und T. Pajdla (2007). "Robust Rotation and Translation Estimation in Multiview Reconstruction". In: *Conference on Computer Vision and Pattern Recognition*.

Martinez, F. V., J. C. de Pina und J. Soares (2007). "Algorithms for Terminal Steiner Trees". In: *Theoretical Computer Science* 389 (1–2), S. 133–142.

Matas, J., O. Chum, M. Urban und T. Pajdla (2002). "Robust Wide Baseline Stereo from Maximally Stable Extremal Regions". In: *British Machine Vision Conference*.

Mayer, H. (2014). "Efficient Hierarchical Triplet Merging for Camera Pose Estimation". In: *German Conference on Pattern Recognition*.

Mayer, H., J. Bartelsen, H. Hirschmüller und A. Kuhn (2012). "Dense 3D Reconstruction from Wide Baseline Image Sets". In: *Outdoor and Large-Scale Real-World Scene Analysis*. Lecture Notes in Computer Science. Springer-Verlag, S. 285–304.

Mayer, H. und M. Michelini (2016). "Orientierung großer Bildverbände". In: *Handbuch der Geodäsie*. Hrsg. von W. Freeden und R. Rummel. Springer Berlin Heidelberg, S. 197–228.

McGlone, J. C. (2013). *Manual of Photogrammetry*. 6. Auflage.

Mehlhorn, K. (1988). "A Faster Approximation Algorithm for the Steiner Problem in Graphs". In: *Information Processing Letters* 27.3, S. 125–128.

Michel, L. V. (1975). "A Note on Matchings in Graphs". In: *Cahiers du Centre d'Etudes de Recherche Opérationnelle*.

Michelini, M. und H. Mayer (2014). "Detection of Critical Camera Configurations for Structure from Motion". In: *ISPRS International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XL-3/W1, S. 73–78.

– (2016). "Efficient Wide Baseline Structure from Motion". In: *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* III-3, S. 99–106.

Mikolajczyk, K. und C. Schmid (2002). "An Affine Invariant Interest Point Detector". In: *European Conference on Computer Vision*.

– (2005a). "A Performance Evaluation of Local Descriptors". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.10, S. 1615–1630.

– (2005b). "Scale and Affine Invariant Interest Point Detector". In: *International Journal of Computer Vision* 60.1, S. 63–86.

Mishkin, D., J. Matas und M. Perdoch (2015). "MODS: Fast and Robust Method for Two-View Matching". In: *Computer Vision and Image Understanding* 141, S. 81–93.

Mitra, K. und R. Chellappa (2008). "A Scalable Projective Bundle Adjustment Algorithm using the L_∞-Norm". In: *Indian Conference on Computer Vision*.

Morel, J.-M. und G. Yu (2009). "ASIFT: A New Framework for Fully Affine Invariant Image Comparison". In: *SIAM Journal on Imaging Sciences* 2.2, S. 438–469.

Moulon, P., P. Monasse und R. Marlet (2012). "Adaptive Structure from Motion with a Centario Model Estimation". In: *Asian Conference on Computer Vision*.

– (2013). "Global Fusion of Relative Motions for Robust, Accurate and Scalable Structure from Motion". In: *International Conference on Computer Vision*.

Muja, M. und D. G. Lowe (2009). “Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration”. In: *International Conference on Computer Vision Theory and Application*.

Murtagh, F. und P. Contreras (2012). “Algorithms for Hierarchical Clustering: An Overview”. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 2.1, S. 86–97.

Ni, K. und F. Dellaert (2012). “HyperSfM”. In: *International Conference on 3D Imaging, Modeling, Processing, Visualization Transmission*.

Nistér, D. (2000). “Frame Decimation for Structure and Motion”. In: *European Workshop on 3D Structure from Multiple Images of Large-Scale Environments*.

– (2004). “An Efficient Solution to the Five-Point Relative Pose Problem”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.6, S. 756–777.

Nistér, D. und F. Schaffalitzky (2006). “Four Points in Two or Three Calibrated Views: Theory and Practice”. In: *International Journal of Computer Vision* 67.2, S. 211–231.

Nistér, D. und H. Stewenius (2006). “Scalable Recognition with a Vocabulary Tree”. In: *Conference on Computer Vision and Pattern Recognition*.

Oliva, A. und A. Torralba (2001). “Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope”. In: *International Journal of Computer Vision* 42.3, S. 145–175.

Pearson, R. K. (2002). “Outliers in Process Modeling and Identification”. In: *IEEE Transactions on Control Systems Technology* 10.1, S. 55–63.

Philbin, J., O. Chum, M. Isard, J. Sivic und A. Zisserman (2007). “Object Retrieval with Large Vocabularies and Fast Saptial Matching”. In: *Conference on Computer Vision and Pattern Recognition*.

Philbin, J. und A. Zisserman (2008). “Object Mining Using a Matching Graph on Very Large Image Collections”. In: *Indian Conference on Computer Vision, Graphics & Image Processing*.

Prim, R. C. (1957). “Shortest Connection Networks and some Generalizations”. In: *Bell System Technical Journal* 36.6, S. 1389–1401.

Quinlan, J. R. (1996). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc.

Reich, M. und C. Heipke (2016). “Convex Image Orientation from Relative Orientations”. In: *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*. Band III–3. 3, S. 107–114.

Repko, J. und M. Pollefeys (2005). “3D Models from Extended Uncalibrated Video Sequences: Addressing Key-frame Selection and Projective Drift”. In: *International Conference on 3-D Digital Imaging and Modeling*.

Robins, G. und A. Zelikovsky (2005). “Tighter Bounds for Graph Steiner Tree Approximation”. In: *SIAM Journal on Discrete Mathematics* 19.1, S. 122–134.

Rogers, D. J. und T. T. Tanimoto (1960). “A Computer Program for Classifying Plants”. In: *Science* 132.3434, S. 1115–1118.

Roth, L., A. Kuhn und H. Mayer (2017). “Wide-Baseline Image Matching with Projective View Synthesis and Calibrated Geometric Verification”. In: *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science* 85.2, S. 85–95.

Samet, H. (2006). *Foundations of Multidimensional and Metric Data Structures*. Morgan Kaufmann.

Schaffalitzky, F. und A. Zisserman (2002). “Multi-view Matching for Unordered Image Sets, or "How Do I Organize My Holiday Snaps?"”. In: *European Conference on Computer Vision*.

Schönberger, J. L., A. C. Berg und J.-M. Frahm (2015a). “Efficient Two-View Geometry Classification”. In: *German Conference on Pattern Recognition*.

– (2015b). “PAIGE: PAirwise Image Geometry Encoding for Improved Efficiency in Structure-from-Motion”. In: *Conference on Computer Vision and Pattern Recognition*.

Schönberger, J. L. und J.-M. Frahm (2016). “Structure-from-Motion Revisited”. In: *Conference on Computer Vision and Pattern Recognition*.

Schönberger, J. L., T. Price, T. Sattler, J.-M. Frahm und M. Pollefeys (2016). “A Vote-and-Verify Strategy for Fast Spatial Verification in Image Retrieval”. In: *Asian Conference on Computer Vision*.

Shum, H.-Y., Q. Ke und Z. Zhang (1999). “Efficient Bundle Adjustment with Virtual Key Frames: A Hierarcical Approach to Multi-Frame Structure from Motion”. In: *Conference on Computer Vision and Pattern Recognition*.

Silpa-Anan, C. und R. Hartley (2008). “Optimized kD-Trees for Fast Image Descriptor Matching”. In: *Conference on Computer Vision and Pattern Recognition*.

Sivic, J. und A. Zisserman (2003). “Video Google: A Text Retrieval Approach to Object Matching in Videos”. In: *International Conference on Computer Vision*.

Snavely, N., S. M. Seitz und R. Szeliski (2006). “Photo Tourism: Exploring Image Collections in 3D”. In: *SIGGRAPH*.

– (2008). “Skeletal Graphs for Efficient Structure from Motion”. In: *Conference on Computer Vision and Pattern Recognition*.

Steedly, D., I. Essa und F. Dellaert (2003). “Spectral Partitioning for Structure from Motion”. In: *International Conference on Computer Vision*.

Steele, K. L. und P. K. Egbert (2006). “Minimum Spanning Tree Pose Estimation”. In: *International Symposium on 3D Data Processing, Visualization and Transmission*.

Strecha, C., A. Bronstein, M. Bronstein und P. Fua (2012). “LDAHash: Improved Matching with Smaller Descriptors”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.1, S. 66–78.

Sweeney, C., T. Sattler, T. Hollerer, M. Turk und M. Pollefeys (2015). “Optimizing the Viewing Graph for Structure-from-Motion”. In: *International Conference on Computer Vision*.

Szeliski, R. (2011). *Computer Vision: Algorithms and Applications*. Springer.

Thormählen, T., H. Broszio und A. Weissenfeld (2004). “Keyframe Selection for Camera Motion and Structure Estimation from Multiple Views”. In: *European Conference on Computer Vision*.

Toldo, R., R. Gherardi, M. Ferenzena und A. Fusiello (2015). “Hierarchical Structure-and-Motion Recovery from Uncalibrated Images”. In: *Computer Vision and Image Understanding* 140.C, S. 127–143.

Torr, P. H. (1997). “An Assessment of Information Criteria for Motion Model Selection”. In: *Conference on Computer Vision and Pattern Recognition*.

Torr, P. H., A. W. Fitzgibbon und A. Zisserman (1999). “The Problem of Degeneracy in Structure and Motion Recovery from Uncalibrated Image Sequences”. In: *International Journal of Computer Vision* 32.1, S. 27–44.

Torr, P. H. und A. Zisserman (1997). “Robust Parametrization and Computation of the Trifocal Tensor”. In: *Image and Vision Computing* 15, S. 591–605.

– (2000). “MLESAC: A New Robust Estimator with Application to Estimating Image Geometry”. In: *Computer Vision and Image Understanding* 78, S. 138–156.

Torr, P. H., A. Zisserman und S. J. Maybank (1998). “Robust Detection of Degenerate Configurations while Estimating the Fundamental Matrix”. In: *Computer Vision and Image Understanding* 71.3, S. 312–333.

Triggs, B., P. F. McLauchlan, R. Hartley und A. W. Fitzgibbon (1999). “Bundle Adjustment – A Modern Synthesis”. In: *International Conference on Computer Vision*.

Tuytelaars, T. und K. Mikolajczyk (2008). “Local Invariant Feature Detectors: A Survey”. In: *Foundations and Trends in Computer Graphics and Vision* 3.3, S. 177–280.

Warren, M., D. McKinnon, H. He und B. Upcroft (2010). “Unaided Stereo Vision Based Pose Estimation”. In: *Australasian Conference on Robotics and Automation*.

Wefelscheid, C. (2013). "Monocular Camera Path Estimation Cross-linking Images in a Graph Structure". Dissertation. Technische Universität Berlin.

Wilson, K. und N. Snavely (2014). "Robust Global Translations with 1DSfM". In: *European Conference on Computer Vision*.

Winter, P. (1987). "Steiner Problem in Networks: A Survey". In: *Networks* 17 (2), S. 129–167.

Witten, I. H., A. Moffat und T. C. Bell (1999). *Managing Gigabytes: Compressing and Indexing Documents and Images*.

Wu, C. (2012). *SiftGPU: A GPU Implementation of Scale Invariant Feature Transform (SIFT)*.

– (2013). "Towards Linear-Time Incremental Structure from Motion". In: *International Conference on 3D Vision*.

Wu, C., S. Agarwal, B. Curless und S. M. Seitz (2011). "Multicore Bundle Adjustment". In: *Conference on Computer Vision and Pattern Recognition*.

Zach, C., A. Irschara und H. Bischof (2008). "What Can Missing Correspondences Tell Us About 3D Structure and Motion". In: *Conference on Computer Vision and Pattern Recognition*.

Zach, C., M. Klopschitz und M. Pollefeys (2010). "Disambiguating Visual Relations Using Loop Constraints". In: *Conference on Computer Vision and Pattern Recognition*.

Zelikovsky, A. Z. (1993). "An 11/6-approximation Algorithm for the Network Steiner Problem". In: *Algorithmica* 9.5, S. 463–470.

Zhang, J. (2013). "Advancements of Outlier Detection: A Survey". In: *ICST Transactions on Scalable Information Systems* 13.1, S. 1–26.

Zhao, G., L. Chen, G. Chen und J. Yuan (2010). "KPB-SIFT: A Compact Local Feature Descriptor". In: *ACM International Conference on Multimedia Pages*.

Stichwortverzeichnis