

Bachelorarbeit am Institut für Informatik der Freien Universität Berlin,

Arbeitsgruppe: Artificial & Collective Intelligence

# Kamerakalibrierung und (teil-)automatisiertes Stitching

Peter Strümpel

Matrikelnummer: #####

[peter.struempel@fu-berlin.de](mailto:peter.struempel@fu-berlin.de)

1. Gutachter & Betreuer: Prof. Dr. Tim Landgraf

2. Gutachter: Prof. Dr. Raúl Rojas

Berlin, 03.11.2016

## **Zusammenfassung**

Das *BeesBook* Projekt beschäftigt sich mit der Langzeitanalyse des Kommunikationsverhaltens von Bienen. Dazu werden Bienen auf einer Wabe eindeutig identifiziert und ihre Bewegung mit Hilfe mehrerer Kameras erfasst. Zwei hochauflösende Kameras nehmen jeweils eine Wabenseite auf, wobei stets eine Kamera eine Wabenhälfte erfasst. Die Positionen der Bienen werden dann in Relation zum Bild einer Wabenhälfte bestimmt. Für die weitere Analyse ist es erforderlich, dass die Position in Relation zu einem Gesamtbild einer Wabenseite beschrieben werden kann. Diese Arbeit entwickelt daher Lösungen, welche es ermöglichen, die Koordinaten aus den Bildern der Wabenhälften in ein Gesamtkoordinatensystem zu überführen.

## **Eidesstattliche Erklärung**

Ich versichere hiermit an Eides Statt, dass diese Arbeit von niemand anderem als meiner Person verfasst worden ist. Alle verwendeten Hilfsmittel wie Berichte, Bücher, Internetseiten oder ähnliches sind im Literaturverzeichnis angegeben, Zitate aus fremden Arbeiten sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungskommission vorgelegt und auch nicht veröffentlicht.

03.11.2016

Peter Strümpel

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Das BeesBook Projekt . . . . .	1
1.2	Motivation . . . . .	1
1.3	Aufbau der Arbeit . . . . .	2
<b>2</b>	<b>Stand der Forschung</b>	<b>3</b>
<b>3</b>	<b>Implementierung</b>	<b>5</b>
3.1	Anforderungen . . . . .	5
3.1.1	Allgemeine Anforderungen . . . . .	5
3.1.2	Besondere Anforderungen . . . . .	6
3.2	Kamerakalibrierung . . . . .	7
3.2.1	Modell der Lochkamera . . . . .	7
3.2.2	Verzeichnungen . . . . .	8
3.2.3	Bestimmung der Verzeichnungsparameter . . . . .	10
3.3	Stitching . . . . .	11
<b>4</b>	<b>Evaluierung</b>	<b>16</b>
4.1	Genauigkeit des Stitchings . . . . .	16
4.1.1	Feature-based Stitching . . . . .	16
4.1.2	Rectangle Stitching . . . . .	18
4.2	Laufzeit . . . . .	19
<b>5</b>	<b>Diskussion und Ausblick</b>	<b>20</b>
<b>A</b>	<b>Anhang</b>	<b>21</b>

# 1 Einleitung

## 1.1 Das BeesBook Projekt

Das *BeesBook* Projekt der AG *Artificial & collective Intelligence* befasst sich mit den Möglichkeiten zur Langzeitanalyse des Kommunikationsverhaltens von Bienen. Das sich daraus ergebende System zur Analyse kann in drei aufeinander aufbauende Bereiche gegliedert werden:

**I Bilderfassung und Tanzdetektierung.** Ein automatisch funktionierendes Aufnahmesystem erstellt einerseits hochauflösende Bilder einer Bienenwabe und andererseits detektiert es in Echtzeit das Auftreten des sog. *Schwänzeltanzes*. Der Schwänzeltanz ist eine Kommunikationsform der Bienen zur Lagebeschreibung von Futterquellen.

**II Bildanalyse.** Mit Hilfe von Software des maschinellen Sehens werden die Bienen, welche durch Markierungen eindeutig unterscheidbar sind, erkannt, lokalisiert und identifiziert. Die Software der Bildanalyse wird nachfolgend auch als *Pipeline* bezeichnet.

**III Postprocessing.** Die erhobenen Daten werden für verschiedene Auswertungen, wie zum Beispiel das Erkennen von Folgebienen (passive Kommunikationspartner eines Schwänzeltanzes) oder von Trophalaxis genutzt.

[1]

## 1.2 Motivation

Während der Bilderfassung in **I** wird eine Bienenwabe von 4 hochauflösenden Kameras erfasst. Zwei der Kameras nehmen dabei eine Wabenseite auf, wobei immer eine der beiden Kameras die linke oder die rechte Wabenhälfte erfasst (vgl. **Abb. 1.1**). Die Pipeline aus Schritt **II** wird dann mit den Bildern aus **einer** Kamera gespeist und die Positionen der Bienen werden anhand ihrer Lage in dem Bild in Pixeln beschrieben. Die gewonnenen Koordinaten stehen in keiner direkten Relation zur gesamten Wabe.

Ziel dieser Arbeit ist es, eine Lösung zu präsentieren, welche die bildbezogenen Koordinaten in ein einheitliches auf die gesamte Wabe bezogenes Koordinatensystem überführt. Es wird gewissermaßen ein Panorama der Koordinaten aus dem linken und rechten Bild erzeugt. Das Erstellen eines Gesamtbildes aus mehreren Einzelbildern wird im Allgemeinen als **Stitching** bezeichnet. Nachfolgend wird der Begriff aber auch für das Zusammenfügen einzelner Bildkoordinatensysteme und den darin lokalisierten Koordinaten verwendet.

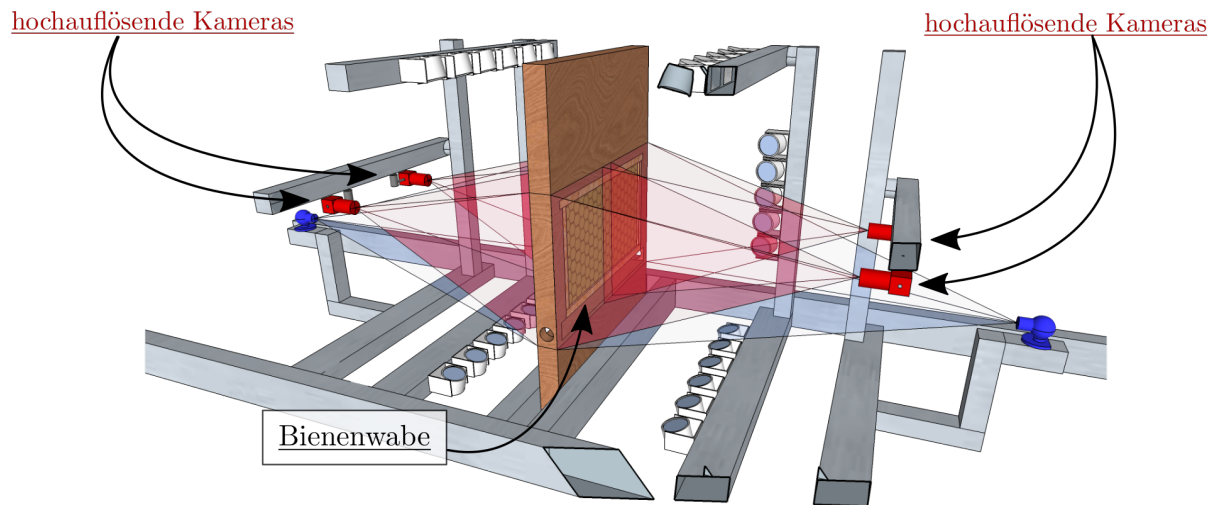


Abbildung 1.1: Die Bienenwabe wird von beiden Seiten, jeweils von zwei hochauflösenden Kameras erfasst. Jede der hochauflösenden Kameras erfasst eine Hälfte der Wabe [2].

### 1.3 Aufbau der Arbeit

Diese Arbeit unterteilt sich in fünf weitere Bereiche, deren Inhalte im Folgenden kurz zusammengefasst werden:

**Abschn. 2: Stand der Forschung.** Dieser Abschnitt stellt einen kurzen Überblick über bereits existierende Methoden des Stitchings von Bildern dar.

**Abschn. 3: Implementierung.** Hier wird zunächst die Problemstellung des Stitchings genauer erläutert und anschließend auf die Grundlagen der Kamerakalibrierung eingegangen. Schlussendlich werden die an das BeesBook Projekt angepassten Stitching-Prozesse vorgestellt.

**Abschn. 4: Evaluierung.** Dieser Part dient der Bewertung der auf Grundlage von **Abschn. 3** implementierten Software.

**Abschn. 5: Diskussion und Ausblick.** In diesem Teil wird darauf eingegangen, inwiefern eine Problemlösung erreicht werden konnte und in welchen Bereichen Verbesserungen möglich sind.

## 2 Stand der Forschung

Das Hauptziel des Stitchings von Bildern ist es, ein möglichst fugenloses Panorama aus sich überlappenden Einzelbildern zu erzeugen. Um dies zu erreichen, existieren mehrere Techniken und Ansätze. Diese versuchen auf verschiedene Art und Weise die Bilder in ein Gesamtbild zu überführen und Übergänge zwischen den Bildern zu verbergen.

Eine der ältesten und weit verbreitetsten Stitching-Methoden besteht dabei in dem Auffinden einer globalen Homographie anhand von Feature Matching [3] - [5]. Dazu werden sich entsprechende Punkte im Überlappungsbereich der zu stichenden Bildern bestimmt, welche dann über eine Homographie ineinander überführt werden. Der Vorteil einer globalen Homographie besteht darin, dass nach einer Überführung eines Bildes in ein anderes die geometrischen Eigenschaften erhalten bleiben. Ist die betrachtete Szene jedoch nicht plan, treten im Überlappungsbereich sog. Parallaxen (siehe [Abschn. 3.1.2](#)) auf.

Neuere Methoden versuchen aufbauend auf einer initialen globalen Homographie durch lokale Verzerrungen, sowie mit geschickter Platzierung des Saumes (Bereich, an dem die Bilder zusammengefügt werden), einen Übergang zu schaffen, in welchem die Parallaxen nicht mehr sichtbar sind. Das vordergründige Ziel der Methoden ist es, ein 'ästhetisches' Gesamtbild zu erzeugen. Dies geschieht jedoch auf Kosten der geometrischen Eigenschaften, welche durch die lokalen Verzerrungen und nicht linearen Abbildungen zerstört werden. Daher sind diese Methoden nicht auf unsere Problemstellung anwendbar. Für das Stitching der Koordinaten und der darauffolgenden Auswertung durch das Postprocessing ist es essentiell, dass diese erhalten bleiben. Trotz dessen seien diese Arbeiten hier vollständigkeithalber und um einen Einblick in die Thematik des Stitchings zu geben, kurz vorgestellt.

Eine der ersten Arbeiten, die von der Verwendung lediglich einer Homographie abweicht, findet sich in Gao *et al.* [6]. In dieser wird die betrachtete Szene in eine Distanz-Ebene und eine Grund-Ebene unterteilt. Für beide Ebenen wird dann jeweils eine Homographie bestimmt, welche gewichtet zu einer als duale Homographie bezeichneten nicht linearen Transformation zusammengefügt werden. Dieser Ansatz scheitert jedoch, wenn relativ große Strukturen existieren, die keiner der beiden Ebenen zugeordnet werden können und bedingt dadurch das Auftreten von Parallaxen.

Eine jüngere Arbeit entwickelte eine Methode, welche eine leicht variierende *affine* Abbildung für das Bild Stitching verwendet [7]. Dabei wird eine globale affine Transformation an mehreren Stellen des Überlappungsbereiches so verzerrt, dass

das Auftreten von Artefakten minimiert wird. Zaragoza *et al.* [8] stellen eine Verbesserung dieser Arbeit vor, indem eine sanft variierende *projektive* Abbildung benutzt wird, die aufgrund der größeren Anzahl an Freiheitsgraden eine bessere Ausrichtung der Bilder ermöglicht. Der Nachteil der in [8] vorgestellten Technik liegt jedoch darin, dass die projektive Transformation auch in Bereiche, welche nicht in dem Überlappungsbereich liegen, extrapoliert wird. Dies führt somit zu starken perspektivischen Verzerrungen in Bereichen, welche weiter vom Überlappungsbereich entfernt sind. Chang *et al.* [9], sowie Lin *et al.* [10] kombinieren diesen Ansatz daher mit einer Ähnlichkeitstransformation. Lediglich der Bereich innerhalb der Überlappung wird über eine variierende projektive Abbildung transformiert und der Bereich, welcher außerhalb der Überlappung liegt, wird über eine Ähnlichkeitstransformation ausgerichtet.

## 3 Implementierung

Im folgenden Abschnitt wird zunächst die Aufgabenstellung dokumentiert und dann erläutert, wie die erarbeiteten Lösungen für die Problemstellung sind.

### 3.1 Anforderungen

#### 3.1.1 Allgemeine Anforderungen

Innerhalb der Pipeline werden die Koordinaten der Bientags in Bildkoordinaten erfasst und ausgegeben. Die Bildkoordinaten haben dabei ihren Koordinatenursprung immer in der linken oberen Ecke des Bildes, unabhängig von der eigentlichen Orientierung der aufnehmenden Kamera zur Wabe. Eine Wabenseite wird von zwei Kameras erfasst, wobei entweder die linke oder rechte Seite der Wabe aufgenommen wird. Vordergründiges Ziel der Arbeit ist es, die Koordinaten aus der linken und rechten Abbildung der Wabe in ein einheitliches Koordinatensystem zu überführen. Hierbei ist wichtig zu betonen, dass es *nicht* um das 'Stitching' der Bilder

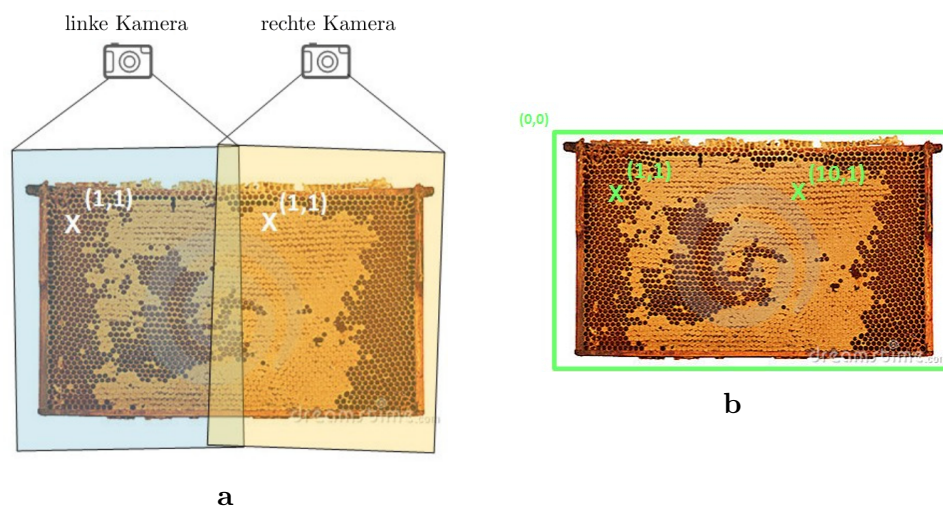


Abbildung 3.1: (a) Schematische Darstellung des Ist-Zustandes. Zwei Kameras erfassen jeweils eine Hälfte der Wabe, wobei die Koordinaten bzgl. des Bildes erfasst werden. (b) Gewünschte Koordinaten-Darstellung. Die Koordinaten werden in einem Koordinatensystem dargestellt.

geht, sondern um das der Koordinaten. Die Bilder der linken und rechten Wabenhälfte aus den Vorjahren des Projekts wurden nicht zeitgleich aufgenommen, was auch einen Fokus auf das Stitching der Bilder wenig sinnvoll macht. Hinzu kommt, dass beim Stitching der Bilder mit einem höheren Rechenaufwand zu rechnen ist, als wenn lediglich die für die weitere Auswertung notwendigen Punktkoordinaten der Bientags gestitched werden.



### 3.1.2 Besondere Anforderungen

Aufgrund des Aufnahmesetups ergeben sich dabei verschiedene Punkte, die berücksichtigt werden müssen:

**Verzeichnungen.** Die Linseneigenschaften der Kamera können zu einer Verzerrung der Bilder führen und müssen korrigiert werden.

**Variables Aufnahmesetup.** Die Position der Kamera zur Wabe wurde seit Beginn der Aufnahmen mehrmals verändert.

**Zeitlicher Versatz.** Da die Bilder nicht gleichzeitig aufgenommen wurden, ist ein feature-based Stitching nur bedingt auf den unbearbeiteten Bildern möglich.

**Automatisch.** Aufgrund der großen (Bild-) Datenmenge des Projekts soll das Stitching möglichst automatisch funktionieren.

**Unterschiedliche Kamerazentren/Unterschiedliche Tiefen.** Eine projektive Transformation (Homographie) zwischen zwei Bildern existiert nur dann, wenn die Bilder mit oder ohne Rotation von einem gleichen Kamerastandpunkt aufgenommen werden oder die aufgenommene Szene plan ist. Trifft dies nicht zu, hat dies zur Folge, dass Bereiche, die als Vorder- und Hintergrund klassifiziert werden, nicht sauber aufeinander abgebildet werden können (siehe Parallaxe) [6, S. 50] [7, S. 1] [11, S. 312].

**Parallaxe.** Eine  $P$ . bezeichnet die scheinbare Änderung der Position eines Objektes, wenn sich die Aufnahmeposition ändert. Dies hat zur Folge, dass beispielsweise eine Biene, welche nicht plan mit der Wabe ist, auf dem Bild der linken Kamera im Verhältnis zum Untergrund weiter rechts erscheint, als sie es auf dem Bild der rechten Kamera tut (vgl. Abb. 3.2).

**Geringer Überlappungsbereich.** Die Überschneidung beider Bilder ist recht klein, was in der Kombination mit der dort auftretenden Parallaxe eine Verringerung von validen Merkmalen, welche für das Stitching benötigt werden, zur Folge hat.

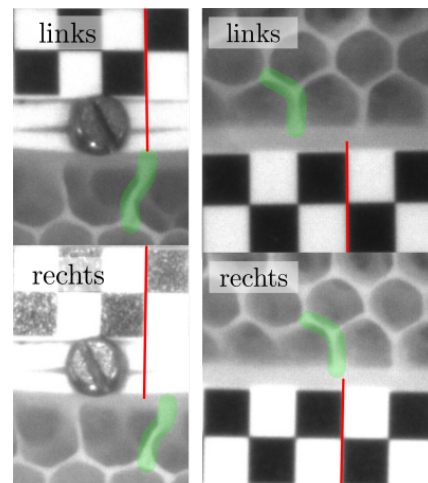


Abbildung 3.2: Veranschaulichung der Parallaxe, die Wabe und der Rahmen der Wabe (Schachbrettmuser) besitzen eine unterschiedliche Tiefe und erscheinen dadurch versetzt zueinander auf Bildern, welche von der linken oder rechten Kamera aufgenommen wurden (Quadratlänge = 4mm).

## 3.2 Kamerakalibrierung

Bevor das eigentliche Stitching und die Messungen auf den Fotos durchgeführt werden können, ist es zunächst notwendig, die unter [Abschn. 3.1.2](#) erwähnten Verzerrungen zu beseitigen. Um dies zu erreichen, muss das Kamerasystem auf seine Eigenschaften hin untersucht werden, da diese Einfluss auf das Bild und damit auch auf die darauf ausgeführten Messungen haben. Die Bestimmung dieser Abbildungsparameter wird als Kamerakalibrierung bezeichnet [11, S. 339] [12, S. 153]. Um diese Parameter bestimmen zu können, muss das Kamerasystem zunächst als ein sog. *Kameramodell*, welches eine sinnvolle  $3D \rightarrow 2D$  Abbildung darstellt, abstrahiert werden [11, S. 293].

### 3.2.1 Modell der Lochkamera

Wir werden uns im Folgenden auf das Modell der sogenannten *Lochkamera* beschränken, da dies unseren Anwendungsfall am besten beschreibt und erweitern dieses durch die Modellierung der Linsenverzerrungen. Im *Lochkameramodell* werden Weltpunkte mittels einer Zentralprojektion auf Bildpunkte abgebildet. Dabei liegen Weltpunkte und Bildpunkte auf einem Strahl, welcher vom Kamerazentrum ausgeht. Das Kamerazentrum liegt im

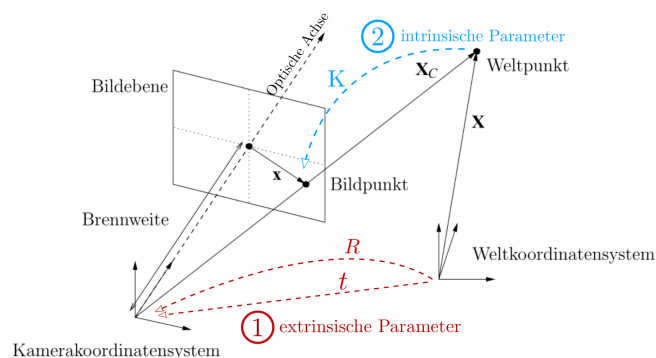


Abbildung 3.3: Veranschaulichung des Lochkameramodells.

Ursprung des Koordinatensystems der Kamera [13, S. 13]. Das Modell der Lochkamera kann dabei vereinfacht durch eine Projektionsmatrix  $P$ , welche einen Weltpunkt  $\mathbf{X}$  in einen Bildpunkt  $\mathbf{x}$  überführt, beschrieben werden:

$$\mathbf{x} = \mathbf{X} \cdot P$$

Die Projektionsmatrix kann weiter in eine Kalibriermatrix  $K$ , eine Rotationsmatrix  $R$  und einen Translationsvektor  $\mathbf{t}$  zerlegt werden, sodass:

$$P = K \cdot (R|\mathbf{t}) \quad (3.1)$$

$$\Rightarrow \mathbf{x} = K \cdot (R|\mathbf{t}) \cdot \mathbf{X} \quad (3.2)$$

Ein Weltpunkt  $X$  wird also zuerst durch die sog. *extrinsischen Kameraparameter* ( $R|\mathbf{t}$ ) ins Kamerakoordinatensystem überführt und von dort über die auch als *intrinsischen Kameraparameter* bezeichnete Kalibriermatrix  $K$  auf die Bildebene abgebildet (vgl. Abb. 3.3).

### 3.2.2 Verzeichnungen

Das Modell der Lochkamera geht, wie in der Abb. 3.3 gezeigt, davon aus, dass eine lineare Beziehung zwischen Welt- und Bildpunkt existiert, d.h. es wird davon ausgegangen, dass Weltpunkt  $X$ , Bildpunkt  $x$  und das Kamerazentrum  $C$  auf demselben optischen Strahl liegen. Aufgrund der Linseneigenschaften des Objektivs und der Lage des Kamerasensors zur Linse kommt es bei realen Kameras jedoch zu Verzeichnungen.

**Radiale Verzeichnung.** Auf Fotos lässt sich meist verstärkt zum Rand hin eine Verzeichnung von Geraden erkennen. So werden Geraden nicht wie in der Zentralprojektion ebenfalls auf Geraden abgebildet, sondern auf gekrümmte Linien [11, S. 357]. Dieses Phänomen wird als *radiale Verzeichnung* bezeichnet und nimmt zu, je weiter der betrachtete Bildpunkt von der Bildmitte entfernt ist (vgl. Abb. 3.4).

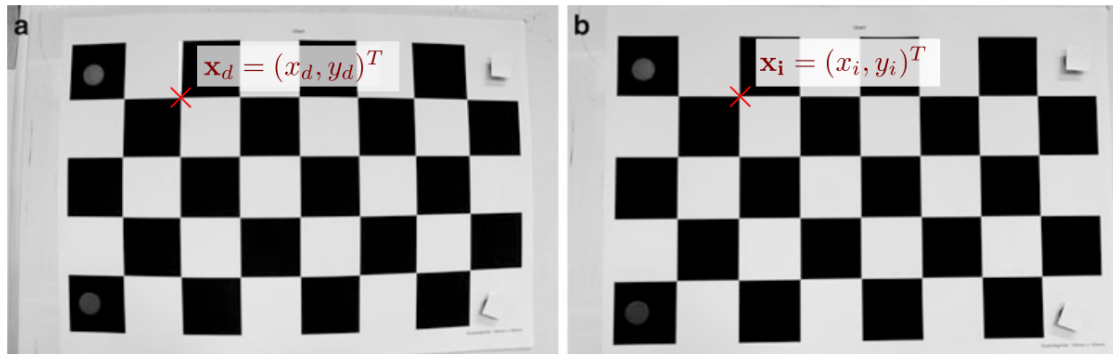


Abbildung 3.4: (a) radiale Verzeichnung, (b) korrigierte Verzeichnung [11, S. 357].

Sei  $\mathbf{x}_i = (x_i, y_i)^T$  ein idealer linearer Bildpunkt, d.h. ein ohne Verzeichnung auf die Bildebene abgebildeter Punkt und  $\mathbf{x}_d = (x_d, y_d)$  der reale verzerrte Punkt. Dann kann die *radiale Verzeichnung* näherungsweise beschrieben werden durch:

$$\begin{pmatrix} x_d \\ y_d \end{pmatrix} = \begin{pmatrix} x_i \\ y_i \end{pmatrix} \cdot (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \quad (3.3)$$

mit  $r^2 = x_i^2 + y_i^2$ .

$k_1, k_2$  und  $k_3$  werden dabei als *Koeffizienten der radialen Verzeichnung* bezeichnet [14, S. 2].

**Tangentiale Verzeichnung.** Falls die Linse und die Bildebene nicht perfekt parallel zueinander sind, kann dies ebenfalls zu Verzeichnungen führen, welche als *tangentiale Verzeichnungen* bezeichnet werden (vgl. Abb. 3.5).

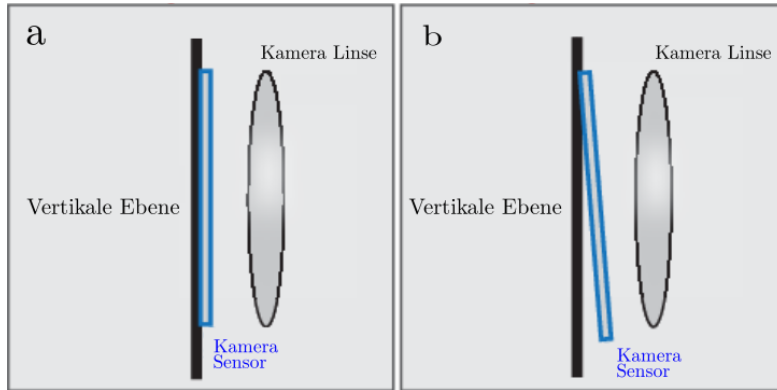


Abbildung 3.5: (a) Keine *tangentiale Verzeichnung*, da *Linse* und *Sensor* parallel sind, (b) *tangentiale Verzeichnung*, da *Linse* und *Sensor* nicht parallel sind [15].

Die *tangentiale Verzeichnung* kann durch folgende Gleichung:

$$\begin{pmatrix} x_d \\ y_d \end{pmatrix} = \begin{pmatrix} x_i \\ y_i \end{pmatrix} + \begin{pmatrix} 2p_1x_iy_i & + & p_2(r^2 + 2x_i^2) \\ p_1(r^2 + 2y_i^2) & + & 2p_2x_iy_i \end{pmatrix} \quad (3.4)$$

mit  $r^2 = x_i^2 + y_i^2$

approximiert werden.  $p_1$  und  $p_2$  heißen *Koeffizienten der tangentiaalen Verzeichnung* [14, S. 2].

### 3.2.3 Bestimmung der Verzeichnungsparameter

Zur Bestimmung der Verzeichnungsparameter wird auf eine bereits in Matlab implementierte Lösung zurückgegriffen. Diese beruht auf der Kamerakalibrierung nach Zhang [16] und gliedert sich grob in folgende Schritte:

#### Kalibrierungsprozess

Ziel: Bestimmung der Verzeichnungsparameter der Kamera.

Ablauf:

1. Erstellung von **Fotos eines ebenen Kalibriermusters**, welches genug besondere Merkmale enthält und bei welchem ausreichend metrische Eigenschaften bekannt sind. (In unserem Fall wurde ein Schachbrettmuster gewählt, wobei die besonderen Merkmale die Ecken der Quadrate sind.)
2. **Extrahieren der besonderen Merkmale** aus den Fotos.
3. **Bestimmung der extrinsischen und intrinsischen Parameter** der verwendeten Kamera. Dies geschieht unter der Annahme, dass keine Verzeichnungen existieren. Mit Hilfe der Bildkoordinaten der Merkmale und dem Wissen über deren relative Lage zueinander in Weltkoordinaten, werden die gesuchten Parameter bestimmt.
4. Gleichzeitige Verbesserung und **Bestimmung aller Parameter**, d.h. auch die der möglichen Verzeichnungen. Dies wird durch die Verwendung des Levenberg-Marquardt-Algorithmus durchgeführt, sodass der *Reprojection Error* möglichst gering wird. Der Reprojection Error beschreibt den Abstand zwischen einem auf dem Kalibriermuster erkannten Punkt und dem ihm entsprechenden Weltpunkt, welcher zurück in die Ebene projiziert wird.

### 3.3 Stitching

Um die Koordinaten aus den Bildern der linken und rechten Hälfte der Wabe in ein Koordinatensystem zu überführen, müssen zunächst die Abbildungsvorschriften gefunden werden, welche einen Punkt in das einheitliche Koordinatensystem überführen. Um diese zu bestimmen, werden zuerst Bilder, welche unter dem gleichen Kamerasetup entstanden sind, gestitcht. Aus diesem Prozess werden dann die nötigen Parameter für das Stitching der Koordinaten ermittelt.

Es wurden zwei Lösungswege entwickelt. Der Erste adressiert vor 2016 erstellte Bilder und basiert auf den in [Abschn. 2](#) vorgestellten Techniken. Dieser setzt für ein bestmögliches Stitching, das Auffinden einer großen Anzahl an in einer Ebene liegenden Features voraus. Jedoch ist dies aufgrund der in [Abschn. 3.1.2](#) benannten Anforderungen erschwert. Aus diesem Grund wurde für neuere Bilder ein Ansatz entwickelt, in welchem auf dem Wabenrand markierte Punkte auf ein abstrahiertes Rechteck abgebildet werden. Beide Ansätze werden im Folgenden dargestellt.

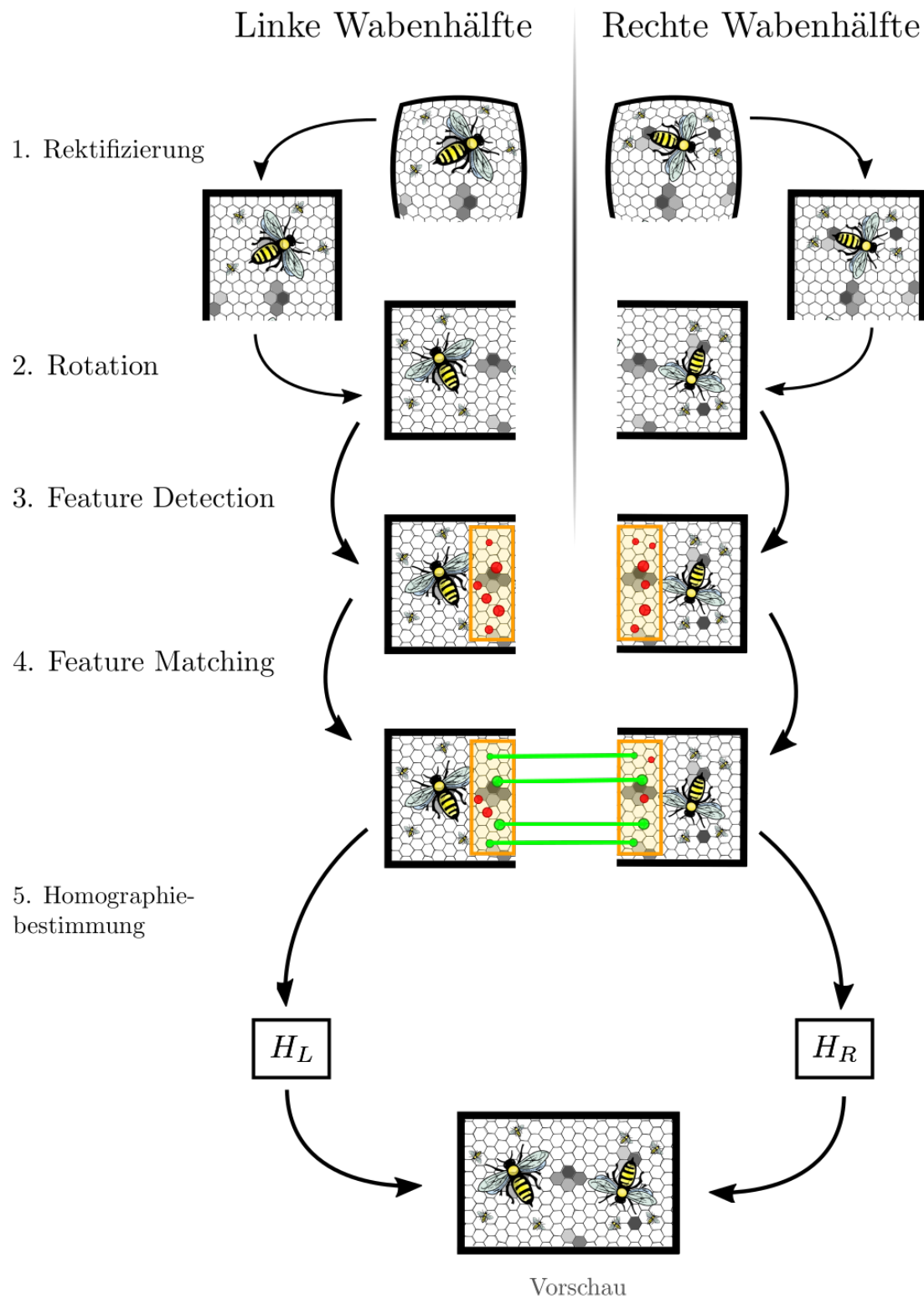
1. Ansatz:

## Feature-based Stitching Pipeline

Ziel: Bestimmung der Homographien, welche Koordinaten aus dem linken und dem rechten Bild in ein einheitliches Koordinatensystem überführen. Basierend auf Feature Matching.

Ablauf: (Für ein besseres Verständnis bitte die [Abb. 3.6](#) auf Seite 13 beachten.)

1. **Rektifizierung.** Anhand der in der Kalibrierung bestimmten Parameter werden die Verzeichnungen korrigiert (siehe [Abschn. 3.2](#)).
2. **Rotation.** Die Bilder werden so rotiert, dass die Wabenmitte jedes Bildes einer Wabenhälfte zueinander zeigt, sodass ein einfacheres Auffinden der Transformationsmatrix möglich ist.
3. **Feature Detection.** Auf jedem Bild werden in einem eingeschränkten Bereich *Keypoints*, auch als Features bezeichnet, und deren *Deskriptoren* erfasst. Der Suchbereich wird einerseits horizontal auf den Überlappungsbereich eingeschränkt und andererseits vertikal so verkleinert, dass der Rahmen der Wabe nicht erfasst wird. Der Grund liegt darin, dass aus mehreren Tests hervorging, dass aus dem weißen Rand, zu viele ähnliche Features extrahiert wurden, welche das nachfolgende Matching zu stark negativ beeinflusst haben.
4. **Feature Matching.** Den Keypoints des einen Bildes werden anhand ihrer Deskriptoren Keypoints des anderen Bildes zugeordnet. Die Keypoints beschreiben die Lage eines Merkmals und ihre Deskriptoren den näheren Bereich des Keypoints.
5. **Bestimmung der Transformationsmatrix/Homographie.** Mit Hilfe der Punktpaare wird eine Homographie bestimmt, welche die rechten Bildkoordinaten in Bildkoordinaten der linken Hälfte abbildet. Hier ist es möglich, zwischen projektiver, affiner, ähnlicher und euklidischer Transformationen zu wählen [[17](#), S. 44].

Abbildung 3.6: *Feature-based Stitching Pipeline* (vgl. Seite 12).



2. Ansatz:

## Rectangle Stitching Pipeline

Ziel: Bestimmung der Homographien, welche Koordinaten aus dem linken und dem rechten Bild in ein einheitliches Koordinatensystem überführen. Basierend auf einer Abbildung von markanten Punkten auf ein Rechteck.

Voraussetzung: Der Rahmen der Bienenwabe wurde mit markanten Punkten (Ecken in einem Schachbrettmuster-Streifen) versehen. Wir bezeichnen sie vereinfacht als *Schachpunkte*. Dieser Streifen ist so befestigt, dass jeder Punkt auf dem Streifen einen ihm im rechten Winkel zum Rand gegenüberliegenden korrespondierenden Schachpunkt besitzt (vgl. [Abb. 3.7](#)).

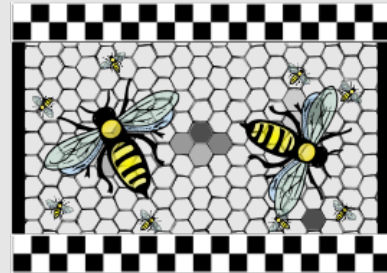
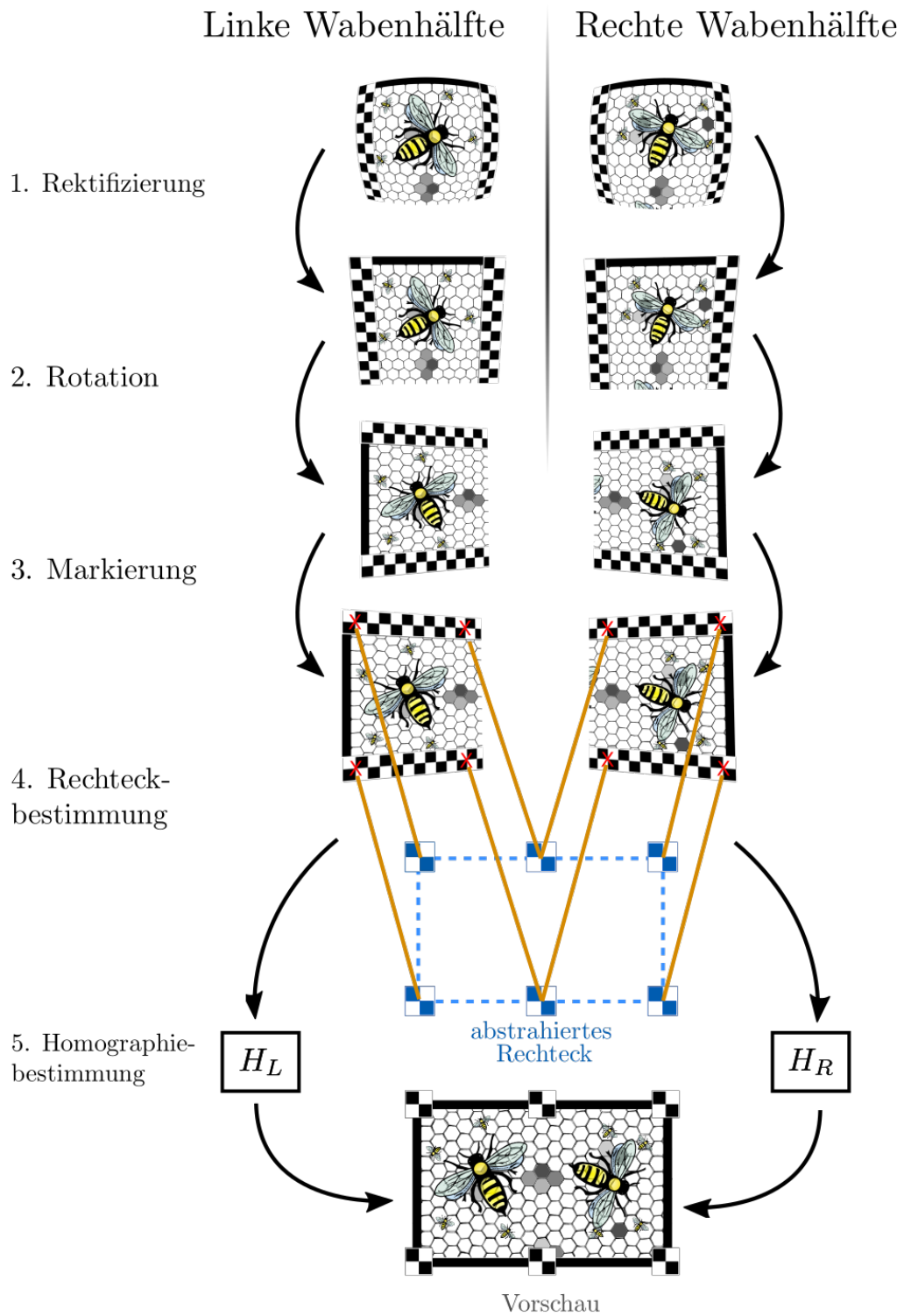


Abbildung 3.7: Bienenwabe mit den Schachbrettmuster-Streifen.

Ablauf: (Für ein besseres Verständnis bitte die [Abb. 3.8](#) auf Seite 15 beachten.)

1. **Rektifizierung.** Anhand der in der Kalibrierung bestimmten Parameter werden die Verzeichnungen korrigiert (siehe [Abschn. 3.2](#)).
2. **Rotation.** Die Bilder werden so rotiert, dass die Wabenmitte jedes Bildes einer Wabenhälfte zueinander zeigt, sodass ein einfacheres Auffinden der Transformationsmatrix möglich ist.
3. **Markierung.** Auf jedem Bild einer Wabenhälfte werden manuell vier Schachpunkte, die ein Rechteck bilden, markiert. Zwei der Punkte müssen im Überlappungsbereich liegen und auch in dem Bild der anderen Wabenhälfte markiert werden.
4. **Rechteckbestimmung.** Anhand des Wissens über die Lage der markierten Schachpunkte in Weltkoordinaten und ihrer relativen Lage zueinander wird ein Rechteck bestimmt, auf welchem die insgesamt 8 Schachpunkte liegen sollen.
5. **Bestimmung der Transformationsmatrix/Homographie.** Es wird für beide Bilder einer Wabenseite jeweils eine Homographie bestimmt, welche die 4 markierten Punkte in das Rechteck überführt.

Abbildung 3.8: *Rectangle Stitching* (vgl. Seite 14).

## 4 Evaluierung

Im folgendem Abschnitt wird analysiert, wie gut die in [Abschn. 3](#) vorgestellten Stitching-Prozeduren dem gewünschten Ziel der Bildung eines einheitlichen Koordinatensystems gerecht werden.

### 4.1 Genauigkeit des Stitchings

Um die Qualität des Stitchings zu überprüfen, wurden 25 Bildpaare aus dem Jahr 2016 zufällig ausgewählt. In den Bildern der linken und rechten Wabenhälfte wurden dann durchschnittlich 36 sich entsprechende Punkte ausgewählt und abgespeichert. Anhand der in [Abschn. 3](#) vorgestellten Techniken wurden dann die Parameter für das Stitching der Koordinaten bestimmt. Anschließend wurden die markierten korrespondierenden Punkte auf ein Gesamtkoordinatensystem abgebildet und der euklidische Abstand zwischen den Punktpaaren bestimmt.

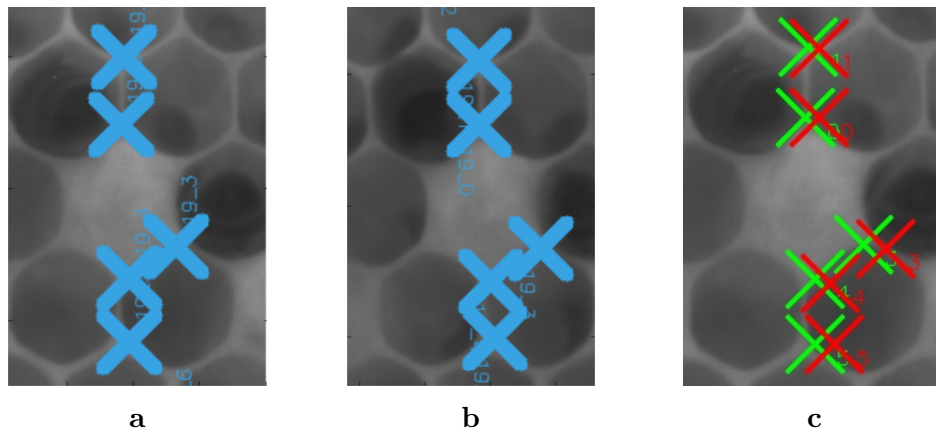


Abbildung 4.1: (a) Markierte Punkte auf dem Bild der linken Wabenhälfte. (b) Markierte Punkte auf dem Bild der rechten Wabenhälfte. (c) Korrespondierende Punkte abgebildet auf ein Gesamt-panorama. Die grünen Markierungen entsprechen den Punkten aus [a](#) und die roten denen aus [b](#).

#### 4.1.1 Feature-based Stitching

Innerhalb der Feature-based Stitching Pipeline ist es möglich, verschiedene Homographien mit sich unterscheidenden Freiheitsgraden zu bestimmen. Wir werden uns nachfolgend auf die ähnliche Transformation beschränken, da sie einerseits die besseren Ergebnisse im Vergleich zu den Transformationen mit mehr Freiheitsgraden liefert und andererseits die Differenz zu den Ergebnissen der euklidischen Transformation recht gering ausfällt.

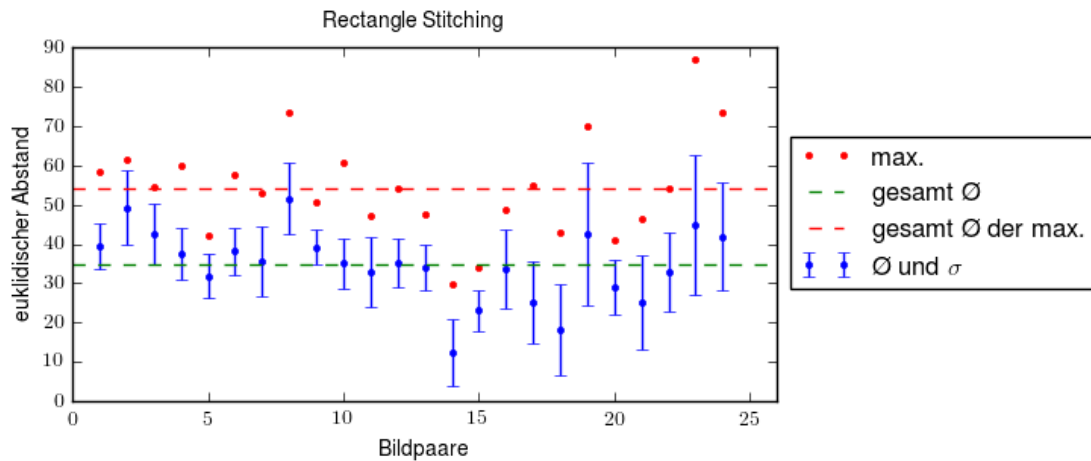


Abbildung 4.2: Die **blauen Fehlerbalken** markieren den **durchschnittlichen euklidischen Abstand**  $\bar{\sigma}$  und die **Standardabweichung**  $\sigma$  korrespondierender Punkte über mehrere Bildpaare. Die **grüne Linie** gibt den Gesamtdurchschnitt des Abstandes über alle Bildpaare hinweg an. Die **roten Punkte** geben den maximalen euklidischen Abstand eines Punktpaares im Bildpaar an. Die **rote Linie** markiert wiederum deren Durchschnitt.

Als Gesamtdurchschnitt des euklidischen Abstands über alle betrachteten Bildpaare erhalten wir einen Wert von 34,63  $px$ . Dies ist in Anbetracht, dass eine Bienenwabenzelle auf den Bildern in etwa einen Durchmesser zwischen 75 und 90  $px$  besitzt, ein gutes Ergebnis. Auch unter Beachtung der Standardabweichung, welche in Abb. 4.2 veranschaulicht ist, scheint die Streuung innerhalb eines Bildpaares über alle Paare recht gering zu sein. Der Gesamtdurchschnitt der Standardabweichungen beträgt 8,98  $px$ . Wichtig für die Beurteilung der Qualität ist noch, den *worst case* für die Abbildung eines Punktes zu betrachten. Daher wurde für jedes Bildpaar jeweils das Punktpaar mit dem größten euklidischen Abstand bestimmt, in Abb. 4.2 als rote Punkte dargestellt. Der Durchschnitt dieser Abstände ist als rote Linie eingezeichnet und beträgt rund 54,32  $px$ .

Daraus folgt, dass beim Stitching von Koordinaten (aus unterschiedlichen Wabenhälften) im Überlappungsbereich für eine Biene zwei Koordinatenpunkte existieren. Diese können nach den ermittelten Werten einen maximalen Abstand von 54,32  $px$  zueinander besitzen. Infolgedessen muss das Postprocessing und Tracking der Bienen so angepasst werden, dass innerhalb des Überlappungsbereiches die Doppelungen erkannt und verarbeitet werden können. Die ermittelten Werte können dabei als Referenz für einen Suchradius dienen, sodass die Doppelung nur in einem bestimmten Umfeld zum Bientag gesucht werden.

### 4.1.2 Rectangle Stitching

Um einen direkten Vergleich zu ermöglichen, wurden für das Rectangle Stitching die gleichen Bildpaare und markierten Punkte wie für das Feature-based Stitching gewählt.

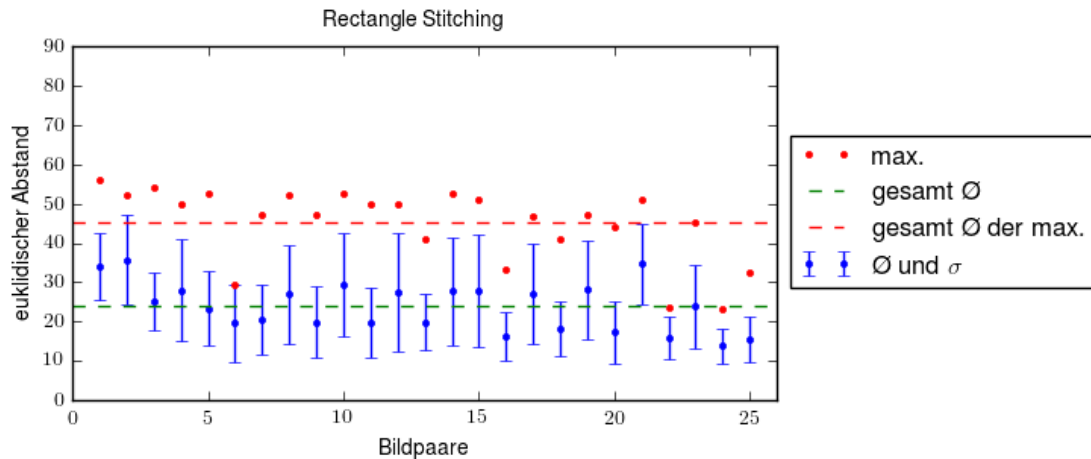


Abbildung 4.3: Die **blauen Fehlerbalken** markieren den **durchschnittlichen euklidischen Abstand**  $\bar{\varnothing}$  und die **Standardabweichung**  $\sigma$  korrespondierender Punkte über mehrere Bildpaare. Die **grüne Linie** gibt den Gesamtdurchschnitt des Abstandes über alle Bildpaare hinweg an. Die **roten Punkte** geben den maximalen euklidischen Abstand eines Punktpaares im Bildpaar an. Die **rote Linie** markiert wiederum deren Durchschnitt.

Wie bereits in der **Abb. 4.3** ersichtlich, sind die Lage- und Streuungsmaße des Rectangle Stitchings besser als die des Feature-based Stitching. So ist der Durchschnitt des euklidischen Abstandes über alle Bildpaare um 32,27 % geringer und liegt bei 23,80 *px*. Bzgl. des Durchschnitts über die pro Bildpaar ermittelten maximalen euklidischen Abstände ergibt sich ein Wert von 45,06 *px*, was um 17,05 % geringer ist als der für das Feature-based Stitching ermittelte Wert. Lediglich die Standardabweichung ist mit 9,83 *px* unwesentlich größer. Eine direkte Gegenüberstellung und Zusammenfassung der Ergebnisse beider Verfahren findet sich in **Tab. 1** wieder.

	Rectangle Stitiching	Feature-based Stitching
$\bar{\varnothing}$ aller Abstände	23,80 <i>px</i>	34,63 <i>px</i>
$\bar{\varnothing}$ der max. Abstände	45,06 <i>px</i>	54,32 <i>px</i>
$\bar{\varnothing}$ der Standardabweichung	9,83 <i>px</i>	8,98 <i>px</i>

Tabelle 1: Zusammenfassung und Gegenüberstellung der berechneten Lage- und Streuungsmaße, bzgl. des euklidischen Abstandes über alle Bildpaare der Stitching Verfahren.

## 4.2 Laufzeit

Die durchschnittlichen Laufzeiten der Stitching-Prozesse wurden mit einem System der folgenden Spezifikation bestimmt:

Betriebssystem: Linux Mint 18 64-bit,  
Linux-Kernel: 4.4.0-31-generic,  
Prozessor: Intel<sup>©</sup> Core<sup>™</sup> i5-4200 CPU @ 1.60GHz x 2,  
Arbeitsspeicher: 8 GB.

Als Eingabe und Datengrundlage für das Stitching dienten die gleichen Daten wie in [Abschn. 4.1](#). Die durchschnittliche Laufzeit für die Bestimmung der Stitching Parameter und die Erstellung eines Vorschaubildes anhand des Feature-based Stitching beträgt in etwa 10,42 s. Für das Rectangle Stitching ist hier nur eine Abschätzung von 5 *min* möglich, da hier eine manuelle Eingabe durch den Benutzer erfolgen muss. Das Stitching bzw. Abbilden der Koordinaten auf das Gesamtkoordinatensystem anhand der bestimmten Parameter beträgt für 100 Punkte ungefähr 2,03 *ms*.

## 5 Diskussion und Ausblick

Innerhalb der Arbeit wurden zwei Ansätze aufgezeigt, mit welchen eine Überführung der Koordinaten aus den Bildern der beiden Wabenhälften in ein Gesamtkoordinatensystem möglich ist. Beide Prozesse besitzen, auch wenn sie hinreichende Lösungen der Problemstellung sind, Verbesserungspotential.

Innerhalb der Feaured-Based Pipeline werden die Homographien allein anhand der Feaures und ihrer Deskriptoren bestimmt. Wir können aber zudem davon ausgehen, dass die sich gegenüberliegenden Kanten ausreichend parallel zueinander sind. Zu überprüfen wäre hier, ob diese Information in den Bestimmungsprozess der Homographien miteinbezogen werden kann, sodass eine bessere automatische Bewertung der Qualität der Transformation möglich ist. Eine grobe Idee wäre hier beispielsweise, zunächst die obere und untere Kante auf beiden Bildern zu bestimmen und die Bilder so zu transformieren, dass sie parallel zueinander sind. Anschließend könnten die Bilder zunächst so aneinander ausgerichtet werden, dass sich entsprechende Kanten auf einer Geraden liegen, woraufhin die Bilder so lange ineinander 'geschoben' werden, bis der Abstand zwischen den Features am geringsten ist oder bis die vertikalen Kanten einen bestimmten Abstand zueinander aufweisen.

Für das Rectangle Stitching ist zusätzlich eine vollständige Automatisierung angestrebt, sodass die für das Stitching benötigten markierten Punkte nicht manuell gewählt werden müssen.

Unabhängig davon sollten, um ermittelte Koordinaten auch über einen längeren Zeitraum in Relation zueinander zu setzen, ein allgemeingültiger Nullpunkt, eine Abszissen- und Ordinatenachse definiert werden. Als Nullpunkt wäre hier die linke obere Ecke denkbar und die Koordinatenachsen könnten so gewählt werden, dass sie entlang der Rahmenkanten verlaufen. Aufbauend darauf wäre dann nur noch die Überführung in ein metrisches System ausstehend.

## A Anhang

**RANSAC** (**random sample consensus**) ist ein Algorithmus zur Schätzung eines Modells innerhalb einer Reihe von Messwerten  $S$ , welche Ausreißer und grobe Fehler enthält.

Algorithmus:

- (i) Es werden zufällig  $s$  Punkte aus der Menge der Messwerte  $S$  ausgewählt und anhand dieser ein Modell  $M$  bestimmt.
- (ii) Bestimmung der Menge an Punkten  $S_i$ , welche nur bis zu einem bestimmten Schwellwert  $t$  von dem gewählten Modell  $M$  abweichen. Die Menge  $S_i$  wird als 'consensus' Menge des gewählten Modells  $M$  bezeichnet und definiert die 'Inliers' von  $S$ .
- (iii) Wenn die Größe von  $S_i$  (Anzahl an Inliers) größer als ein festgelegter Schwellwert  $T$  ist, wird das Modell nochmal mit den Punkten aus  $S_i$  geschätzt und der Prozess beendet.
- (iv) Ist die Größe von  $S_i$  kleiner als  $T$ , wird eine neue Teilmenge  $s$  gewählt und die vorherigen Schritte wiederholt.
- (v) Nach  $N$  Schritten wird die größte Consensus Menge  $S_i$  gewählt und das Modell anhand dieser Menge neu geschätzt.

[17, S. 118]



## Literatur

- [1] F. Wario, B. Wild, M. J. Couvillon, R. Rojas, and T. Landgraf, “Automatic methods for long-term tracking and the detection and decoding of communication dances in honeybees,” *Frontiers in Ecology and Evolution*, vol. 3, no. September, pp. 1–14, 2015. [Online]. Available: <http://journal.frontiersin.org/article/10.3389/fevo.2015.00103>
- [2] F. Wario, “private communication,” 2016.
- [3] R. Szeliski, “Image Alignment and Stitching: A Tutorial,” Tech. Rep., oct 2004. [Online]. Available: <https://www.microsoft.com/en-us/research/publication/image-alignment-and-stitching-a-tutorial/>
- [4] M. Brown and D. G. Lowe, “Automatic panoramic image stitching using invariant features,” in *International Journal of Computer Vision*, vol. 74, no. 1, 2007, pp. 59–73. [Online]. Available: <http://matthewalunbrown.com/papers/ijcv2007.pdf>
- [5] R. Szeliski, *Computer vision: algorithms and applications*, ser. Texts in computer science, D. Gries and F. B. Schneider, Eds. London ; New York: Springer, 2011.
- [6] J. Gao, S. J. Kim, and M. S. Brown, “Constructing image panoramas using dual-homography warping,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011, pp. 49–56. [Online]. Available: <https://www.comp.nus.edu/~brown/pdf/cvpr-dualhomography2011.pdf>
- [7] W. Y. Lin, S. Liu, Y. Matsushita, T. T. Ng, and L. F. Cheong, “Smoothly varying affine stitching,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011, pp. 345–352. [Online]. Available: [https://www.researchgate.net/publication/221364663\\_Smoothly\\_varying\\_affine\\_stitching](https://www.researchgate.net/publication/221364663_Smoothly_varying_affine_stitching)
- [8] J. Zaragoza, T. J. Chin, Q. H. Tran, M. S. Brown, and D. Suter, “As-projective-as-possible image stitching with moving DLT,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1285–1298, 2014. [Online]. Available: [http://www.cv-foundation.org/openaccess/content\\_cvpr\\_2013/papers/Zaragoza\\_As-Projective-As-Possible-Image-Stitching\\_2013-CVPR-paper.pdf](http://www.cv-foundation.org/openaccess/content_cvpr_2013/papers/Zaragoza_As-Projective-As-Possible-Image-Stitching_2013-CVPR-paper.pdf)
- [9] C. H. Chang, Y. Sato, and Y. Y. Chuang, “Shape-preserving half-projective warps for image stitching,” in *Proceedings of the IEEE Computer Society*

- Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3254–3261. [Online]. Available: <http://www.csie.ntu.edu.tw/~cyy/publications/papers/Chang2014SPH.pdf>
- [10] C. C. Lin, S. U. Pankanti, K. N. Ramamurthy, and A. Y. Aravkin, “Adaptive as-natural-as-possible image stitching,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June, 2015, pp. 1155–1163. [Online]. Available: [http://www.cv-foundation.org/openaccess/content\\_cvpr\\_2015/papers/Lin\\_Adaptive\\_As-Natural-As-Possible\\_Image\\_2015\\_CVPR\\_paper.pdf](http://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Lin_Adaptive_As-Natural-As-Possible_Image_2015_CVPR_paper.pdf)
- [11] H. Süße and E. Rodner, *Bildverarbeitung und Objekterkennung: Computer Vision in Industrie und Medizin*. Wiesbaden: Springer Vieweg, 2014.
- [12] R. Godding, “6 - Geometric Calibration of Digital Imaging Systems,” in *Computer Vision and Applications*, B. Jähne and H. H. Secker, Eds. San Diego: Academic Press, 2000, pp. 153–175. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/B9780123797773500078>
- [13] S. Rahmann and H. Burkhardt, “Praktikumsversuch: Kamerakalibrierung,” 2011. [Online]. Available: [http://lmb.informatik.uni-freiburg.de/lectures/praktika\\_brox/bvpraktikum/BVAnl\\_kam\\_kalib.pdf](http://lmb.informatik.uni-freiburg.de/lectures/praktika_brox/bvpraktikum/BVAnl_kam_kalib.pdf)
- [14] J. Heikkila and O. Silven, “A four-step camera calibration procedure with implicit image correction,” *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1106–1112, 1997. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=609468>
- [15] “What Is Camera Calibration? - MATLAB and Simulink - MathWorks Deutschland.” [Online]. Available: <https://de.mathworks.com/help/vision/ug/camera-calibration.html>
- [16] F. Zhang and F. Liu, “Parallax-tolerant image stitching,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014. [Online]. Available: <http://web.cecs.pdx.edu/~fliu/papers/cvpr2014-stitching.pdf>
- [17] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, 2nd ed. Cambridge, UK ; New York: Cambridge University Press, 2003.