

Julius-Maximilians-Universität Würzburg  
Institut für Informatik  
Lehrstuhl für Informatik I  
Effiziente Algorithmen und wissensbasierte Systeme

**Bachelorarbeit**

# **Flusserkennung in Luftbildaufnahmen**

Felix Klesen

Eingereicht am 12. Juli 2017

Betreuer:

Prof. Dr. Alexander Wolff

Prof. Dr. Sabine Storandt

# Zusammenfassung

In der vorliegenden Arbeit werden Verfahren zur Beschreibung und Erkennung von Flussabschnitten auf Luftbildern vorgestellt. Die Identifizierung relevanter Merkmale des Flusses und die Definition eines geeigneten Ähnlichkeitsmaßes auf der Basis dieser Merkmale sind zentraler Bestandteil dieser Arbeit. Es konnte gezeigt werden, dass die Verwendung von Polygonzügen als Merkmalmenge einen Fluss eindeutig charakterisieren. Eine Reduzierung der Polygonzüge auf die Winkel seiner Wendetangenten weist immer noch ausreichend Informationsgehalt auf und reduziert die Laufzeit erheblich. Das entwickelte Verfahren auf der Grundlage von Wendepunkten ist dabei invariant bezüglich vier räumlicher Freiheitsgrade.

# Inhaltsverzeichnis

<b>Zusammenfassung</b>	<b>2</b>
<b>1. Einleitung</b>	<b>5</b>
<b>2. Konzept</b>	<b>6</b>
2.1. Flusserkennung . . . . .	6
2.2. Merkmalsbildung . . . . .	6
2.2.1. Sampling . . . . .	8
2.3. Datenbasis . . . . .	8
2.4. Ähnlichkeitsmaß . . . . .	8
<b>3. Testmethodik</b>	<b>11</b>
3.1. Testumgebung . . . . .	11
3.2. Kalibrierung . . . . .	11
3.2.1. Integritätstest . . . . .	11
3.2.2. Test zur Translationsinvarianz . . . . .	12
3.2.3. Test zur Skalierungsinvarianz . . . . .	12
3.2.4. Test zur Rotationsinvarianz . . . . .	12
3.2.5. Test zur Verzerrungsinvarianz . . . . .	12
3.3. Suchalgorithmus . . . . .	12
3.3.1. Test zur Richtungsinvarianz . . . . .	13
3.3.2. Test zu speziellen partiellen Zuordnungen . . . . .	13
3.3.3. Test zu allgemeinen partiellen Zuordnungen . . . . .	13
3.4. Samplingtests . . . . .	13
3.4.1. Sampleratentests . . . . .	13
3.4.2. Jittertest . . . . .	13
3.4.3. Test zu alternativen Samplings . . . . .	14
3.5. Simulationstest . . . . .	14
<b>4. Realisierung mit richtungsbasiertem Fréchetabstand</b>	<b>15</b>
4.1. Datenaufbereitung . . . . .	15
4.2. Ähnlichkeitsmaß . . . . .	17
4.3. Programmablauf . . . . .	18
4.4. Benchmark . . . . .	18
4.4.1. Kalibrierung . . . . .	19
4.4.2. Suchalgorithmus . . . . .	19
4.4.3. Samplingtests . . . . .	19
4.4.4. Simulationstest . . . . .	22

4.5. Analyse . . . . .	22
<b>5. Realisierung mit Wendepunkten</b>	<b>23</b>
5.1. Datenaufbereitung . . . . .	23
5.2. Ähnlichkeitsmaß . . . . .	24
5.3. Programmablauf . . . . .	24
5.4. Benchmark . . . . .	25
5.4.1. Kalibrierung . . . . .	25
5.4.2. Suchalgorithmus . . . . .	26
5.4.3. Samplingtests . . . . .	26
5.4.4. Simulationstest . . . . .	27
<b>6. Fazit</b>	<b>35</b>
<b>7. Ausblick</b>	<b>36</b>
<b>A. Anhang: Testdaten</b>	<b>40</b>

# 1. Einleitung

In einer Zeit, in der das Wissen um die eigene Position durch globale Navigationssatellitensysteme wie z.B. GPS allgemein zugänglich ist, gerät leicht in Vergessenheit, wie anfällig solche Systeme sowohl für technische Probleme, als auch für Missbrauch sind. Daher ist der Bedarf für autonome Lokalisierungssysteme bei kritischen Anwendungen aktueller denn je.

In der Luftfahrt beispielsweise wird heute nur noch selten mit Flugkarten nach Sicht navigiert. Diese Methode ist nicht nur ungenau, sondern setzt teilweise auch eine erhebliche Ortskenntnis voraus, die man von global eingesetzten Piloten kaum erwarten kann. Fällt das genutzte Satellitennavigationssystem aus, hat ein Piloten nur noch wenige Alternativen: Einerseits die Triangulierung der eigenen Position mit Funkfeuern und andererseits die Fremdlokalisierung durch Radar und Übermittlung der Position per Funk. Allerdings beruhen auch diese beiden Methoden auf externen Systemen, deren Ausfall der Pilot nicht kompensieren kann.

Mit der Trägheitsnavigation existiert zwar bereits ein Konzept zur autonomen Lokalisierung, die entsprechenden Systeme verursachen jedoch hohe Kosten bei Anschaffung und im Unterhalt. Außerdem sind sie schwer und ihr Drift muss beständig korrigiert werden.

Eine autonome Alternative zu den genannten Systemen könnte ein kameragestützter Algorithmus zur Lokalisierung sein. Diese Arbeit untersucht, ob der Verlauf von Flüssen eine eindeutige Positionsbestimmung ermöglicht.

Das Konzept für ein solches System muss folgende Teilaspekte berücksichtigen:

- Die Erkennung von Flussstrukturen in den Bilddaten
- Die Herausarbeitung von Merkmalen, die den Flussabschnitt eindeutig charakterisieren
- Das Suchen des dazu passenden Flusses in einer Datenstruktur

Im Folgenden wird dieses Konzept kurz detaillierter vorgestellt. Diese Arbeit wird sich allerdings im Wesentlichen auf die Frage beschränken, wie diese Suche aussehen könnte und mit welchen Merkmalen diese durchzuführen ist. Eine zentrale Rolle spielt dabei die Entwicklung eines geeigneten Verfahrens zur Bestimmung der Ähnlichkeit von Flussabschnitten.

## 2. Konzept

Bei Luftbildaufnahmen muss zwischen passiven und aktiven Abbildungsverfahren unterschieden werden. Passive Verfahren erzeugen ein Bild des Geländes, indem sie das von ihm reflektierte Sonnenspektrum (meist optisch) mit einer Kamera aufnehmen. Hierbei sind je nach Anwendung sowohl analoge als auch digitale (CCD) Kameras üblich. Aktive Verfahren bestrahlen das Gelände selbst im sensitiven Spektrum des Sensors. Ein übliches Verfahren ist hierbei das Synthetic Aperture Radar, kurz SAR, das durch die Wahl des Radar-Spektrums nicht nur Tageslicht-, sondern auch nahezu Wetterunabhängig funktioniert. Das SAR ist der Kamera somit zwar technisch klar überlegen, es ist aber auch teurer und schwerer.

### 2.1. Flusserkennung

Da sich SAR- und CCD-Bilder in Aufbau und Merkmalen ähnlich sind, können grundsätzlich bei beiden dieselben Konzepte zur Flusserkennung angewendet werden. In einem intuitivem Ansatz können die Ufer anhand ihres Kontrasts mit einem Threshold-Filter und Kantenerkennung identifiziert werden. Dieses Konzept wurde z.B. von Jiang et al. [JFZ<sup>+</sup>14] aufgegriffen, um eine robuste Methode zu entwickeln, die unter anderem auch dünne und flache Gewässer von Störeffekten unterscheiden kann.

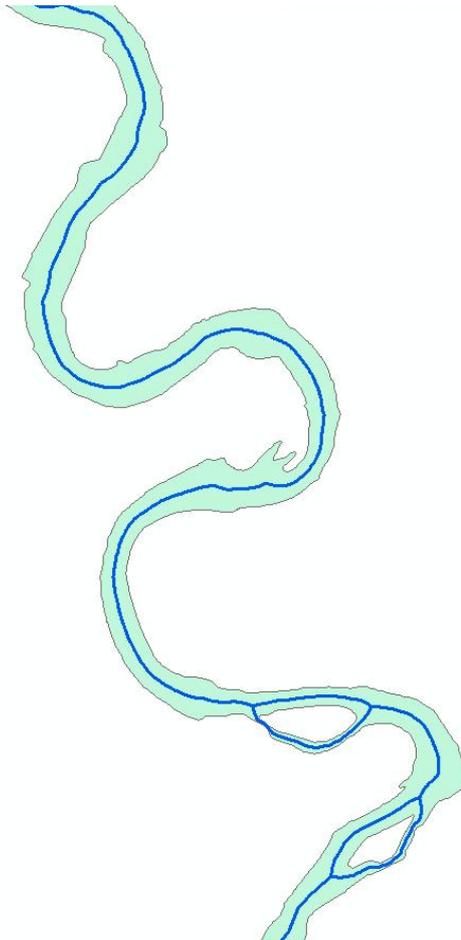
Tian et al. [TWC<sup>+</sup>12] hingegen haben gezeigt, dass eine Flusserkennung auch indirekt über den Hintergrund möglich ist. Dieses Konzept beruht darauf, dass die Umgebung durch seine Komplexität im Vergleich zur Wasseroberfläche reich an Ecken ist und somit die Wahrscheinlichkeit, dass ein Pixel zur Umgebung gehört, steigt, je näher er an einer oder auch mehreren Ecken liegt. Durch gezieltes Training ist es einer Support Vector Machine (SVM) dann möglich, das Bild in Fluss und Umgebung zu unterteilen.

### 2.2. Merkmalbildung

Unabhängig von der gewählten Methode ist das Ergebnis der Flusserkennung, gegebenenfalls nach Bereinigung, ein Flussausschnitt in Form eines Polygons. Die Auflösung des Polygons wird dabei durch die Auflösung des zu Grunde liegenden Bildes definiert. Um bei der späteren Suche nach dem Ausschnitt die Laufzeit zu begrenzen, soll das Polygon soweit vereinfacht werden, dass der Ausschnitt gerade noch eindeutig zugeordnet werden kann. Für die Suche nach Teilstücken eines Flusses ist es erforderlich, dass die in den verwendeten Merkmalen enthaltene Information, an einen definierten Ort gebunden ist. Für Polygone ist diese Bedingung erfüllt, sodass sie beim Vereinfachen lediglich erhalten bleiben muss.

Zur Vereinfachung des Polygons kann z.B. die von Blum [Blu67] definierte mediale Achse verwendet werden. Diese besteht aus allen Punkten innerhalb des Polygons, von denen aus kein eindeutiger, kürzester Weg zum Rand existiert, da mindestens zwei Randpunkte denselben (kürzesten) Abstand aufweisen. Dieser Abstand ist für alle Punkte bekannt, sodass das Polygon daraus wieder verlustfrei generiert werden kann. Verzichtet man auf diese Information erhält man das "topologische Skelett".

Als weitere Vereinfachung können alle kurzen Teilelemente außer Acht gelassen werden, sodass man mit der Hauptachse einen Polygonzug erhält. Die Auflösung des Polygonzugs definiert sich dabei über die Auflösung des verwendeten Polygons. Es soll experimentell gezeigt werden, dass die Reduzierung des Flussausschnitts auf einen Polygonzug für Eindeutigkeit hinreichend ist.



**Abb. 2.1.:** Hauptachse des topologischen Skeletts für einen Fluss. (Quelle: <https://smathermather.com/category/conferences/code-sprint/>)

### 2.2.1. Sampling

Mit dem Begriff Sampling wird in dieser Arbeit die Menge an Punkten bezeichnet, die den Polygonzug beschreibt. Das Sampling spielt sowohl bei der verwendeten Datenbasis als auch bei der Merkmalbildung eine entscheidende Rolle. Die für die Datenbasis genutzte Samplerate (Punkte pro Strecke) ist im Allgemeinen weder konstant, noch bekannt. Daraus ergibt sich eine vom Ähnlichkeitsmaß über alle Polygonzüge und innerhalb der Polygonzüge zu kompensierende Sampling-Range. Ein Ähnlichkeitsmaß muss daher im Rahmen der Sampling-Range unabhängig vom Sampling zu guten Ergebnissen führen, solange der Informationsgehalt, der aus dem Sampling resultiert, für eine eindeutige Zuordnung ausreicht. Ist die Samplerate ausreichend hoch, wird eine weitere Erhöhung den Informationsgehalt des resultierenden Polygonzugs nicht weiter erhöhen. Im umgekehrten Fall, wenn die Samplerate nicht ausreichend ist, ist der Informationsgehalt des Polygonzugs zu niedrig, um den Flussabschnitt eindeutig zu beschreiben.

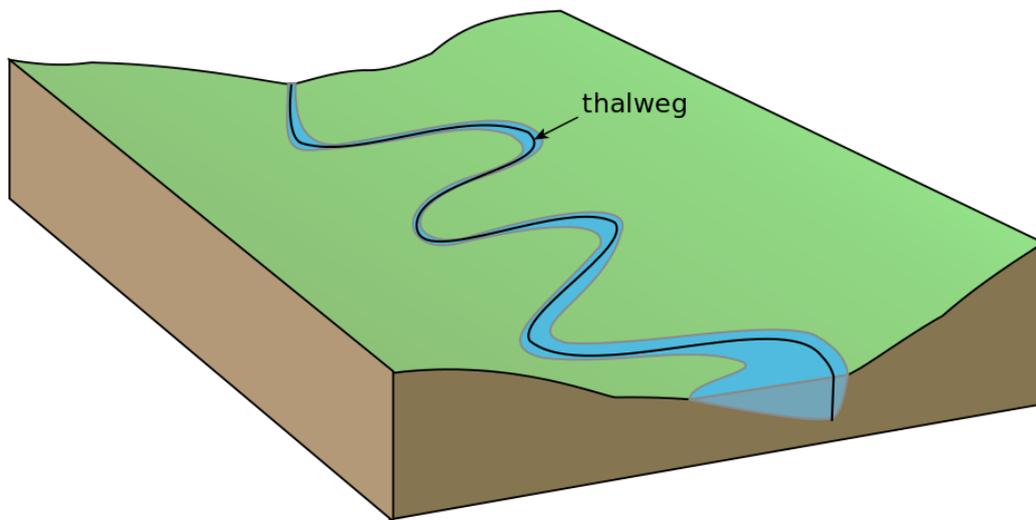
Daraus folgt, dass eine zu geringe Samplerate zu schlechten Ergebnissen führen und eine zu hohe Samplerate folglich gute Ergebnisse liefern wird. Je nachdem welches Ähnlichkeitsmaß verwendet wird, führt eine zu hohe Samplerate jedoch zu längeren Laufzeiten. Für ein optimiertes Laufzeitverhalten bei gleichzeitig guten Ergebnissen muss die Samplerate daher optimal gewählt werden.

## 2.3. Datenbasis

Um den so generierten Polygonzug einem Fluss zuordnen zu können, ist für jeden in Frage kommenden Fluss eine entsprechende Repräsentation als Polygonzug erforderlich. Hierzu bieten sich die Daten der OpenStreetMap (OSM) an, da sie frei zugänglich und flächendeckend verfügbar sind. Weiterhin liegen die Flüsse hier bereits als Polygonzüge vor, wenn auch nicht entlang der medialen Achse, sondern entlang des Talwegs (siehe Abb. 2.2). Unter Anderem dadurch weichen die Repräsentationen desselben Flussabschnitts in Daten und Aufnahme lokal voneinander ab, weisen aber dennoch denselben Verlauf auf. Um diese einander zuzuordnen, ist daher eine geeignete Definition von Ähnlichkeit nötig, unter der die Polygonzüge verglichen werden.

## 2.4. Ähnlichkeitsmaß

In der vorliegenden Problemstellung existieren die Merkmale eines Flusses in unterschiedlichen Koordinatensystemen (Datenbank und Bild). Um eine Ähnlichkeit festzustellen, muss die Transformation zwischen diesen Koordinatensystemen berücksichtigt werden. Diese Transformation besteht aus insgesamt sechs Freiheitsgraden: drei für die Translation entlang der drei Achsen und drei für die Rotation in den drei Ebenen. Für jeden dieser Freiheitsgrade muss entweder die Transformation bekannt sein, oder das Verfahren zur Bestimmung der Ähnlichkeit muss dagegen invariant sein. Je mehr Invarianzen ein solches Verfahren aufweist, desto stärker wird eingeschränkt, welche Arten von Merkmalen noch relevanten Informationsgehalt besitzen. Zusätzlich gilt: Je stärker die Merk-



**Abb. 2.2.:** Talweg eines Flusses. (Quelle: Von Hans Erren - Eigenes Werk, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=26875440>)

malsbildung eingeschränkt wird, desto komplizierter wird auch die Implementierung des Ähnlichkeitsmaßes.

Es gibt bereits diverse Verfahren, die geeignet sind, die Ähnlichkeit von Polygonzügen zu bestimmen. Als Vergleichswert wird dabei das Ähnlichkeitsmaß bestimmt. Im Folgenden werden einige ausgewählte Verfahren vorgestellt.

Der von Fréchet und Heywood [FH12] definierte Fréchetabstand ist eine einfache Möglichkeit, die Ähnlichkeit zweier Funktionen mathematisch zu beschreiben. Sie ist anschaulich die kürzestmögliche Leine, die zwei Individuen verbindet, während diese auf den beiden Funktionen mit jeweils durchgehend positiver Geschwindigkeit propagieren. Der Fréchetabstand besitzt keine Invarianzen, sodass es für seine Bestimmung unabdingbar ist, alle sechs Freiheitsgrade zu kennen.

In der Hausdorff-Metrik [Hau14] wird der (Hausdorff-)Abstand zweier Mengen definiert als das Maximum aller Distanzen zwischen jedem Element beider Mengen und dem jeweils nächstgelegenen Element der anderen Menge. Im Fall zweier Polygonzüge ist der Hausdorffabstand somit die größte Entfernung, die ein Punkt zum ihm nächstgelegenen Punkt der anderen Menge aufweist. Der Hausdorffabstand besitzt ebenfalls keine Invarianzen, sodass auch hier alle 6 Freiheitsgrade bekannt sein müssen.

Abhilfe schafft hier die von Gromov [Gro99] definierte Erweiterung zur Gromov-Hausdorff-Metrik. Diese ist definiert als das Infimum der Hausdorffabstände über alle Einbettungen beider Mengen in einen metrischen Raum. Veranschaulicht bedeutet das, dass die beiden Polygonzüge hinsichtlich des geringsten möglichen Hausdorffabstands optimal verschoben, rotiert und skaliert werden. Somit ist der Gromov-Hausdorffabstand

invariant bezüglich aller 6 Freiheitsgrade.

Bei dem von de Berg und Cook [dBC11] formulierten richtungsbasierten Fréchetabstand (im Folgenden: RbFa) wird analog zum Fréchetabstand für zwei gegebene Polygonzüge jeder Kante des einen Polygonzuges eine oder mehrere Kanten des anderen zugewiesen, sodass die maximale oder durchschnittliche Abweichung der Winkel zwischen den Kanten minimiert wird. Es werden also weder absolute noch relative Positionen benötigt und das Verfahren ist demnach translationsinvariant. Die übrigen drei Freiheitsgrade müssen bekannt sein.

## 3. Testmethodik

In den folgenden Abschnitten werden Testmethoden und ihre Randbedingungen beschrieben, die für die Bewertung von Vergleichssystematiken (Merkmale und Ähnlichkeitsmaß) und deren Implementierung genutzt werden. Die Ähnlichkeit zweier Merkmalsmengen (hier z.B. Polygonzüge) wird dabei als Ähnlichkeitsmaß auf  $\mathbb{R}_0^+$  abgebildet. Je höher das Ergebnis, desto geringer ist die Ähnlichkeit. Daraus ergibt sich, dass das Ähnlichkeitsmaß 0 perfekter Übereinstimmung entspricht.

### 3.1. Testumgebung

Alle in dieser Arbeit beschriebenen Tests wurden in folgender Umgebung durchgeführt:

- Intel Core i7-4702MQ @2,20GHz
- 8GB DDR3 Speicher @1600MHz
- Ubuntu 16.04
- Java 8u121
- Eclipse 3.8

### 3.2. Kalibrierung

Zur Kalibrierung einer Vergleichssystematik müssen alle Invarianzen des Verfahrens mit den zur Verfügung stehenden Merkmalen überprüft werden. Sind alle Invarianzen positiv getestet, erfüllt die Implementierung der Vergleichssystematik das postulierte Verhalten bezüglich der Koordinatensysteme.

#### 3.2.1. Integritätstest

Die Integrität ist innerhalb einer Vergleichssystematik gegeben, wenn eine Merkmalsmenge verglichen mit einer identischen Kopie im selben Koordinatensystem, die bestmögliche Ähnlichkeit aufweist. Das bedeutet, dass das korrekte Ergebnis eines Vergleichs zwischen zwei unterschiedlichen Merkmalsmengen in keinem Fall besser sein darf als das Ergebnis des Vergleichs mit identischen Merkmalsmengen. Das geforderte Ähnlichkeitsmaß für diesen Test ist der Wert 0.

### **3.2.2. Test zur Translationsinvarianz**

Die Translationsinvarianz ist innerhalb einer Vergleichssystematik gegeben, wenn eine Merkmalmenge verglichen mit einer identischen Kopie in einem (in der Ebene) verschobenen Koordinatensystem die bestmögliche Ähnlichkeit aufweist. So wie beim Integritätstest wird auch hier ein Ähnlichkeitsmaß von 0 gefordert. Durch die Forderung von Translationsinvarianz sind absolute Positionen zur Bildung von Merkmalen ungeeignet. Relative Positionen sind jedoch weiterhin nutzbare Informationen.

### **3.2.3. Test zur Skalierungsinvarianz**

Die Skalierungsinvarianz ist innerhalb einer Vergleichssystematik gegeben, wenn eine Merkmalmenge verglichen mit einer identischen Kopie, in einem skalierten (senkrecht gegen die Ebene verschobenen) Koordinatensystem die bestmögliche Ähnlichkeit aufweist. Das geforderte Ähnlichkeitsmaß ist erneut 0. Durch die Forderung von Skalierungsinvarianz sind relative Positionen zur Bildung von Merkmalen ungeeignet. Verhältnisse zwischen relativen Positionen sind jedoch weiterhin nutzbare Informationen.

### **3.2.4. Test zur Rotationsinvarianz**

Die Rotationsinvarianz ist innerhalb einer Vergleichssystematik gegeben, wenn eine Merkmalmenge verglichen mit einer identischen Kopie, in einem (in der Ebene) gedrehten Koordinatensystem die bestmögliche Ähnlichkeit aufweist. Auch hier ist das geforderte Ähnlichkeitsmaß 0. Durch die Forderung von Rotationsinvarianz sind absolute Winkel zur Bildung von Merkmalen ungeeignet. Relative Winkel sind jedoch weiterhin nutzbare Informationen.

### **3.2.5. Test zur Verzerrungsinvarianz**

Die Verzerrungsinvarianz ist innerhalb einer Vergleichssystematik gegeben, wenn eine Merkmalmenge, verglichen mit einer identischen Kopie, in einem (gegen die Ebene) gekippten Koordinatensystem die bestmögliche Ähnlichkeit aufweist. Das geforderte Ähnlichkeitsmaß ist erneut 0. Durch die Forderung von Verzerrungsinvarianz sind relative Winkel zur Bildung von Merkmalen ungeeignet. Verhältnisse zwischen relativen Winkeln sind jedoch weiterhin nutzbare Informationen.

## **3.3. Suchalgorithmus**

Der Suchalgorithmus ist eine Logik, deren Aufgabe darin besteht, einen Flussabschnitt innerhalb einer Datenbank von Flüssen zu finden. Die Merkmale, die den Flussabschnitt und die Flüsse in der Datenbank beschreiben, müssen hierbei in derselben Systematik vorliegen. Die größte Übereinstimmung zwischen dem gesuchten Flussabschnitt und allen Teilstücken aller Flüsse aus der Datenbank ist hierbei das gesuchte Ergebnis.

### **3.3.1. Test zur Richtungsinvarianz**

In der Datenbank wird die Fließrichtung durch die Abfolge der Merkmale von der Quelle zu Senke definiert. Aus Bildinformationen geht die Fließrichtung eines Flusses allerdings normalerweise nicht hervor. Daraus ergibt sich die Forderung, dass der Suchalgorithmus unabhängig von der Fließrichtung das korrekte Ergebnis bestimmen muss. Da das Ähnlichkeitsmaß für die korrekte Richtung gemäß Integritätstest (Abschnitt 3.2.1) 0 ist, wird für die inverse Richtung ein Ähnlichkeitsmaß größer 0 gefordert.

### **3.3.2. Test zu speziellen partiellen Zuordnungen**

Für den Suchalgorithmus wird gefordert, dass ein Teilabschnitt eines Flusses auf der Gesamtlänge desselben Flusses eindeutig identifiziert werden muss. Als Ähnlichkeitsmaß wird der Wert 0 gefordert. Daraus folgt, dass die verwendeten Merkmale eines Flusses in Teilabschnitte unterteilt werden können. Jedes Merkmal muss demnach einem Ort auf dem Fluss zugeordnet sein.

### **3.3.3. Test zu allgemeinen partiellen Zuordnungen**

Für den Suchalgorithmus wird gefordert, dass ein Teilabschnitt eines Flusses auf der Gesamtlänge aller Flüsse aus der Datenbank eindeutig identifiziert werden muss. Als Ähnlichkeitsmaß wird der Wert 0 gefordert.

## **3.4. Samplingtests**

Wie schon in Abschnitt 2.2.1 beschrieben, erfolgt die Erzeugung von Merkmalen durch das Sampeln. Wie ein Merkmal entstanden und wie hoch der Informationsgehalt ist, kann im Allgemeinen nicht bestimmt werden. Von einer Vergleichssystematik wird daher gefordert, dass das Sampling nur geringen Einfluss auf das Ähnlichkeitsmaß hat.

### **3.4.1. Sampleratentests**

Für die Samplerate wird gefordert, dass ein Übersampling keinen signifikanten Einfluss auf das Ähnlichkeitsmaß hat. Hier ist ein Ähnlichkeitsmaß nahe 0 gefordert. Ein Untersampling muss zu einem deutlich schlechteren Ähnlichkeitsmaß führen, da die Samplerate der Flüsse in der Datenbank den erforderlichen Informationsgehalt definiert.

### **3.4.2. Jittertest**

Um sicherzustellen, dass ein Flussabschnitt auch mit Merkmalen gefunden wird, die nicht exakt denen aus der Datenbank entsprechen, muss eine Vergleichssystematik in der Lage sein, eine Unschärfe der Merkmale bis zu einem vertretbaren Maß zu kompensieren. Daraus folgt, dass das Ergebnis im Vergleich unterschiedlicher Merkmalmengen auch dann das korrekte Ergebnis sein kann, wenn es keine absolute Übereinstimmung gibt.

### **3.4.3. Test zu alternativen Samplings**

Für eine Vergleichssystematik wird gefordert, dass unterschiedliche Samplings entlang derselben medialen Achse zu sehr ähnlichen Ergebnissen führen müssen. Dabei wird ein Ähnlichkeitsmaß nahe 0 gefordert. Das bedeutet: Wenn die gleiche Informationsquelle, hier die mediale Achse, genutzt wird, müssen verschiedene Samplings selber Samplerate vergleichbaren Informationsgehalt in den Merkmalen ablegen.

### **3.5. Simulationstest**

Für eine Vergleichssystematik wird gefordert, dass Samplings entlang unterschiedlicher Achsen, die demselben Verlauf folgen, zu ähnlichen Ergebnissen führen müssen. Das bedeutet, dass Merkmale eines Flussabschnitts, die unabhängig von der Datenbank erzeugt wurden, aber einen ähnlichen Informationsgehalt wie dieser Abschnitt in der Datenbank aufweisen, für die Zuordnung dieses Flussabschnitts in der Datenbank geeignet sind. Gefordert wird ein Ähnlichkeitsmaß nahe 0.

Der Simulationstest vereinigt alle vorgestellten Tests in sich und kann daher als Proof of Concept für Vergleichssystematiken genutzt werden.

## 4. Realisierung mit richtungsbasiertem Fréchetabstand

Durch die Problemstellung, die Lokalisierung auf einer Oberfläche, müssen von einem Verfahren zur Bestimmung des Ähnlichkeitsmaßes mindestens zwei Invarianzen gegen Translations-Freiheitsgrade gefordert werden. Daraus ergibt sich, dass der Fréchet-, sowie der Hausdorffabstand, für die Lösung dieses Problems ungeeignet sind. Bietet ein Verfahren weitere Invarianzen, werden weniger zusätzliche Informationen benötigt. Allerdings steigt auch die Komplexität einer Implementierung und der damit verbundene Rechenaufwand.

Unter diesen Voraussetzungen erscheint der richtungsbasierte Fréchetabstand (RbFa) als ein guter Kompromiss und wird daher im Folgenden auf seine Eignung geprüft. Die fehlenden drei Rotations-Invarianzen können z.B. durch Hinzunahme eines Kompasses und eines Accelerometers ausgeglichen werden. Für die weiteren Betrachtungen wird davon ausgegangen, dass die Rotations-Transformation bereits durchgeführt wurde.

### 4.1. Datenaufbereitung

In der OSM-Datenbank werden Flüsse zwar, wie bereits erwähnt, schon als Polygonzüge repräsentiert, allerdings sind sie stark fragmentiert. Die einzelnen Teilstücke variieren dabei in ihrer Länge zwischen wenigen 10 Metern und einigen 10 Kilometern, sodass keine guten Suchbedingungen gegeben sind. Da es sonst nicht möglich wäre, einen Flussausschnitt zu finden, der sich über mehrere Abschnitte erstreckt, ist es erforderlich, die einzelnen Stücke desselben Flusses aneinander zu hängen, sodass für jeden Fluss genau ein Polygonzug im Datenspeicher hinterlegt ist.

Auch wenn diese Anpassung notwendig ist, muss darauf hingewiesen werden, dass die Anpassung abgesehen von Fehlern, die sich aus der OSM-Datenbank fortpflanzen, auch zu konzeptionellen Problemen führen kann. So kann beispielsweise bei einem Fluss, der sich aufgabelt und später wieder zusammenfließt, nur einer der Wege berücksichtigt werden. Das geschieht auch dann, wenn kein eindeutiger Hauptflusslauf erkennbar ist.

Fließt ein Fluss in einen See, wird der aus dem See heraus fließende Fluss von der Datenaufbereitung ignoriert, wenn dieser laut Datenbank derselbe Fluss ist, wie der in den See hinein fließende Fluss.

Die oben genannten Schwächen der Datenaufbereitung sind für das Proof of Concept nicht relevant und werden daher hier nicht korrigiert. Vor einer weitergehenden Nutzung des Systems müssen jedoch Lösungen für diese Schwächen erarbeitet werden.

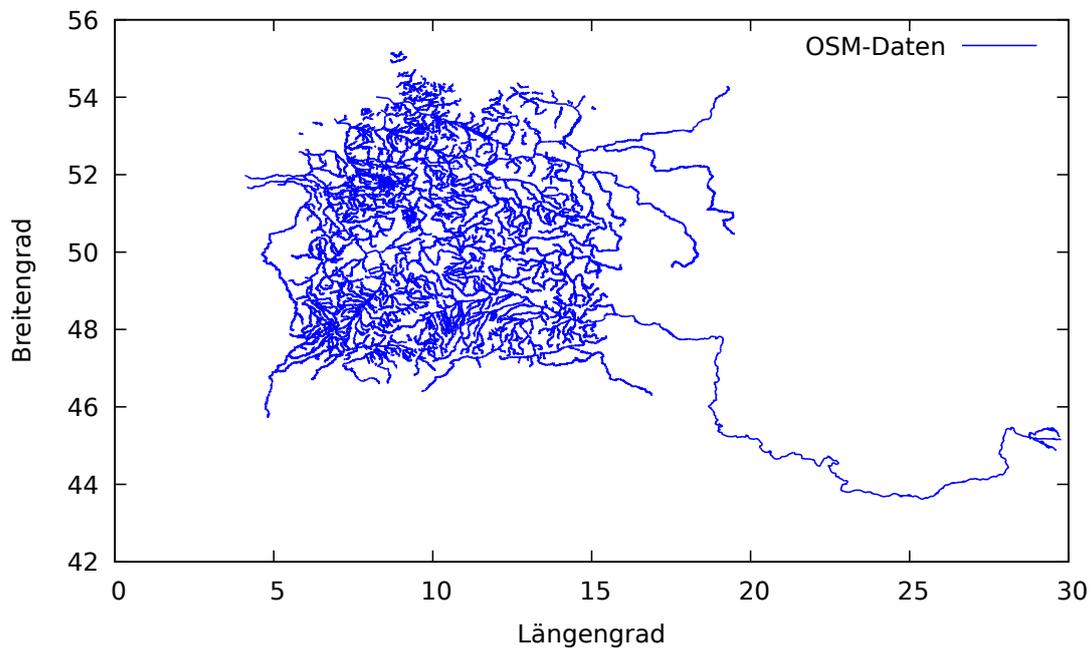
Abgesehen von dieser signifikanten Anpassung wurden bei der Aufbereitung außerdem alle irrelevanten und nicht signifikanten Informationen entfernt. So ist z.B. nicht relevant,

welche Teile eines Flusses schiffbar sind und ein Fluss, der aus weniger als 20 Punkten besteht, ist, wie später gezeigt, nicht signifikant genug. Das Entfernen dieser kurzen Flüsse verbessert nicht nur die Laufzeit, sondern verringert auch die Wahrscheinlichkeit einer falschen Zuordnung.

Die Koordinaten der Daten aus der OSM-Datenbank liegen im WGS 84 (World Geodetic System 1984) vor. Dabei werden Längen- und Breitengrade auf einem Referenzellipsoiden genutzt. Von Bildsystemen erstellte Flussdarstellungen werden jedoch durch eine Linse auf eine Ebene abgebildet. Um einen Vergleich zu ermöglichen, ist es daher erforderlich, die vorliegenden Koordinaten durch eine Projektion ebenfalls auf eine Ebene abzubilden. Aus diesem Grund wurde auf den Koordinaten eine Mercator-Projektion (nach EPSG 3857) durchgeführt.

Der im Folgenden genutzte, grob Deutschland umfassende Datensatz (47,26 bis 55,12 Grad nördliche Breite, 5,91 bis 15,10 Grad östliche Länge) ist ein Kompromiss zwischen dem Anspruch, das Problem global zu lösen, und einer praktikablen Laufzeit für ein Proof of Concept. Der Datensatz besteht nach der Aufbereitung aus 591 Flüssen und 606.059 Knoten. (Knoten/Fluss: Min: 20, Max: 7.807, Median: 717) Gespeichert wurde dieser optimierte Datensatz in einem an xml angelehnten Format, dessen Ziel es ist, einfach lesbar und verständlich zu sein, ohne gleichzeitig einen unnötig hohen Speicheraufwand zu verursachen.

Abbildung 4.1 zeigt die Rohdaten des verwendeten Ausschnitts direkt aus der OSM-Datenbank, während Abbildung 4.2 in grün den gleichen Ausschnitt nach der beschriebenen Aufbereitung zeigt. Nicht nutzbare und damit verlorene Daten sind rot dargestellt.



**Abb. 4.1.:** Flüsse aus der OSM-Datenbank

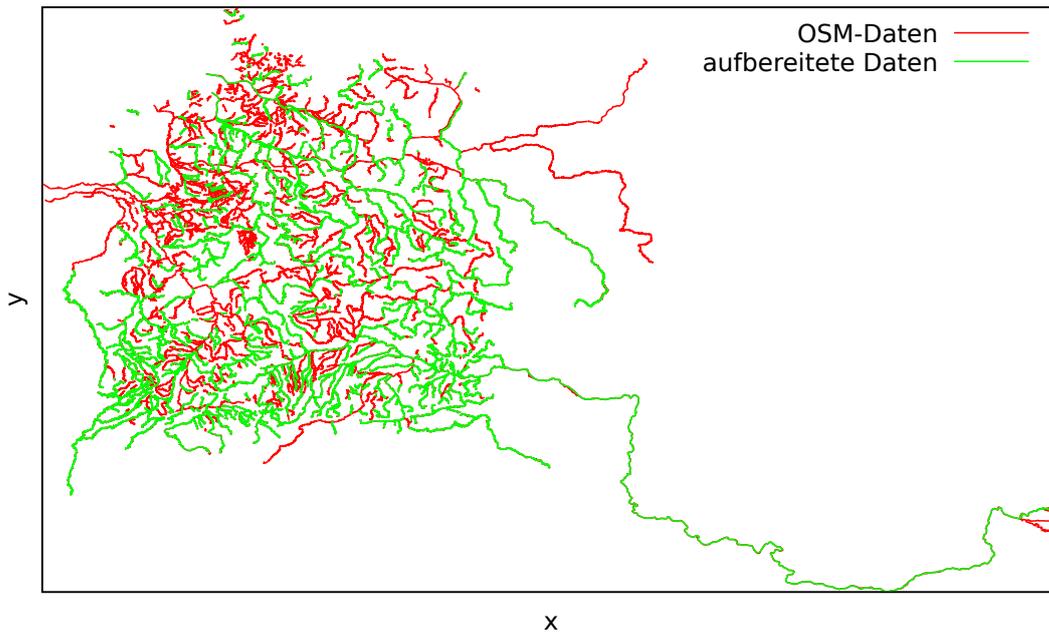


Abb. 4.2.: Flüsse in der aufbereiteten Datenbank

## 4.2. Ähnlichkeitsmaß

Um den RbFa für zwei Polygonzüge zu berechnen, wird das Problem zunächst in seiner Komplexität reduziert, indem ein so genanntes Free-Space-Diagramm (FSD) erstellt wird. Die Achsen entsprechen dabei jeweils strecken-, aber nicht winkelkonform den Polygonzügen. Ein Polygonzug wird demnach so auf die entsprechende Achse abgebildet, dass seine Länge erhalten bleibt, aber alle Ecken ausgeglättet sind. Eine vollständige Zuordnung zwischen den beiden Polygonzügen entspricht in diesem Diagramm also einer Funktion, die die linke untere Ecke mit der oberen rechten Ecke verbindet. Die für unser Problem interessanteren, teilweisen Zuordnungen sind entsprechend Funktionen, die einen beliebigen Punkt auf der x-Achse mit einem beliebigen Punkt höherer x-Koordinate am oberen Rand des FSD verbindet. Aus der Forderung, dass auf beiden Polygonzügen mit durchgehend positiver Geschwindigkeit zu propagieren ist, folgt, dass im FSD nur monotone Funktionen zulässig sind.

Fügt man nun für alle Punkte beider Polygonzüge an den ihnen entsprechenden Stellen beider Achsen orthogonale Linien ein, entstehen Schnittpunkte, Kanten und Zellen, die das Diagramm diskretisieren. Um die Fehler, die durch diese Diskretisierung entstehen, klein zu halten, werden auf den Achsen jeweils  $k$  weitere orthogonale Linien in äquidistanten Abständen eingefügt, die auch wiederum Schnittpunkte bilden, von denen aus Kanten innerhalb der Zellen verlaufen. Ist der Winkel, in dem Kanten relativ zum Koordinatensystem vorliegen dürfen, von unten ( $> 0^\circ$ ) oder von oben ( $< 90^\circ$ ) beschränkt, spricht man von Speed Limits. Dies entspricht einer Einschränkung für das

minimale bzw. maximale Verhältnis der Propagierungsgeschwindigkeiten entlang der Polygonzüge.

Das Gewicht einer Zelle sei nun definiert als der Winkel zwischen den beiden Stücken der jeweiligen Polygonzüge, die sie definieren. Das Gewicht einer Kante wiederum sei ihre euklidische Länge mal das Gewicht der Zelle, die sie durchläuft. Aus der geforderten Monotonie folgt weiter, dass alle Kanten gerichtet sind und auch nur in einer Richtung vorliegen dürfen.

Auf dem so konstruierten gerichteten, gewichteten Graphen entsprechen die gesuchten Funktionen s-t-Wegen und der integrale RbFa lässt sich somit als kürzeste Wege-Problem approximieren. Die Güte der  $(1 + \epsilon)$ -Approximation ergibt sich dabei aus  $\epsilon = 1/k$ . Löst man dieses mithilfe einer Breitensuche, beträgt die Laufzeit  $O(V + E)$ , wobei  $O(V) = n \cdot m/\epsilon$  und  $O(E) = n \cdot m/\epsilon^2$  gilt, sodass für  $\epsilon \leq 1$  gilt, dass  $O(V + E) = O(nm/\epsilon^2)$  ist. Dabei sind  $n$  und  $m$  die Längen der beiden Polygonzüge.

Im Folgenden wird eine 1,1-Approximation ohne Speed Limits verwendet, deren Implementierung von Atlas F. Cook zur Verfügung gestellt wurde. Eine exakte Implementierung wäre mit  $O(nm)$  allerdings um Faktor 100 schneller.

### 4.3. Programmablauf

Eine Eigenschaft des RbFa ist die interne Suche nach partiellen Zuordnungen zwischen den Polygonzügen. Deshalb genügt es, den RbFa für jeden Fluss aus der Datenbank zu bestimmen. Die Iteration über alle möglichen Teilstücke der Flüsse entfällt. Der Programmablauf sieht dabei grob wie folgt aus:

- Einlesen des Datensatzes aus einer Datei in eine im RAM liegende Datenbank
- Zu findenden Polygonzug einlesen
- Über alle Flüsse aus der Datenbank iterieren
  - Besten RbFa zwischen dem Polygonzug und beliebigen Ausschnitten des aktuellen Flusses bestimmen
  - Den gefundenen Flussausschnitt und seinen RbFa merken, falls diese besser ist, als die bisher gemerkte
- Ausgeben der besten Zuordnung

### 4.4. Benchmark

Zur Überprüfung, ob die Implementierung für eine Flusserkennung geeignet ist, werden nachfolgend die in der Testmethodik (Kapitel 3) beschriebenen Tests durchgeführt und bewertet. Dabei ist zu beachten, dass das theoretisch erreichbare Ähnlichkeitsmaß 0 bei einer approximativen Implementierung (wie der hier verwendeten) nicht zwingend erreicht werden muss. Als korrekt gelte das Ergebnis einer Suche, wenn erstens der

richtige Fluss gewählt wurde und zweitens die Anfangs- und Endpunkte um je maximal 10% der Länge abweichen.

#### 4.4.1. Kalibrierung

Da wir von dem RbFa keine Invarianzen bezüglich Rotation und Verzerrung erwarten, können die entsprechenden Tests entfallen. Die Tests zur Integrität sowie zur Translations- und Skalierungsinvarianz führten jeweils erwartungsgemäß zu einem Ähnlichkeitsmaß von 0. Daraus folgt, dass die Kalibrierung der Koordinatensysteme erfolgreich implementiert wurde.

#### 4.4.2. Suchalgorithmus

Der Test zur Richtungsinvarianz bestätigt, dass der RbFa für einen Polygonzug und seine Inverse im Allgemeinen nicht 0 ist. Um bei den nachfolgenden Tests die Hälfte der Laufzeit zu sparen, wird im Folgenden davon ausgegangen, dass die Fließrichtung des Flusses bekannt ist.

Wie bereits im Abschnitt zum Ähnlichkeitsmaß (4.2) beschrieben, wird durch den RbFa automatisch auf partielle Zuordnungen geprüft, sodass der Test zu speziellen partiellen Zuordnungen in diesem Fall keine Aussage über den Suchalgorithmus, sondern über die Implementierung des RbFa trifft. Das Testergebnis bestätigt die korrekte Implementierung.

Der Test zu allgemeinen partiellen Zuordnungen bestätigt die Korrektheit der Implementierung des Suchalgorithmus. Der Test wurde für unterschiedliche Längen mit jeweils 100 Polygonzügen durchgeführt. Die Testergebnisse sind in der ersten Spalte (0) der Tabelle 4.1 dargestellt. Die Tabelle zeigt die Erkennungsraten in Abhängigkeit von der Länge der Flussabschnitte. Aus der Erkennungsrate von 98% für Polygonzüge der Länge 10 ergibt sich, dass Polygonzüge mit weniger als 20 Punkten innerhalb der Datenbank nicht zwingend eindeutig sind. Demnach sind Flüsse mit weniger als 20 Punkten nicht zwingend von Teilabschnitten größerer Flüsse zu unterscheiden. Folgerichtig wurden sie durch die Datenaufbereitung entfernt.

#### 4.4.3. Samplingtests

Im Test zum Untersampling wurde ein Polygonzug mit einer Kopie verglichen, in der jeder zweite Punkt ausgelassen wurde. Somit wurde die Samplerate in diesem Test halbiert. Erwartungsgemäß war das Ähnlichkeitsmaß größer 0, sodass Testpolygonzüge teilweise nicht mehr korrekt erkannt wurden.

Für den Test zum Übersampling wurde ein Polygonzug mit einer Kopie verglichen, in der zwischen allen aufeinander folgenden Punkten ein zusätzlicher Punkt in der euklidischen Mitte eingefügt wurde. Somit wurde die Samplerate in diesem Test verdoppelt. Erwartungsgemäß wurde das Ähnlichkeitsmaß 0 weiterhin erreicht, sodass Testpolygonzüge weiterhin korrekt erkannt wurden.

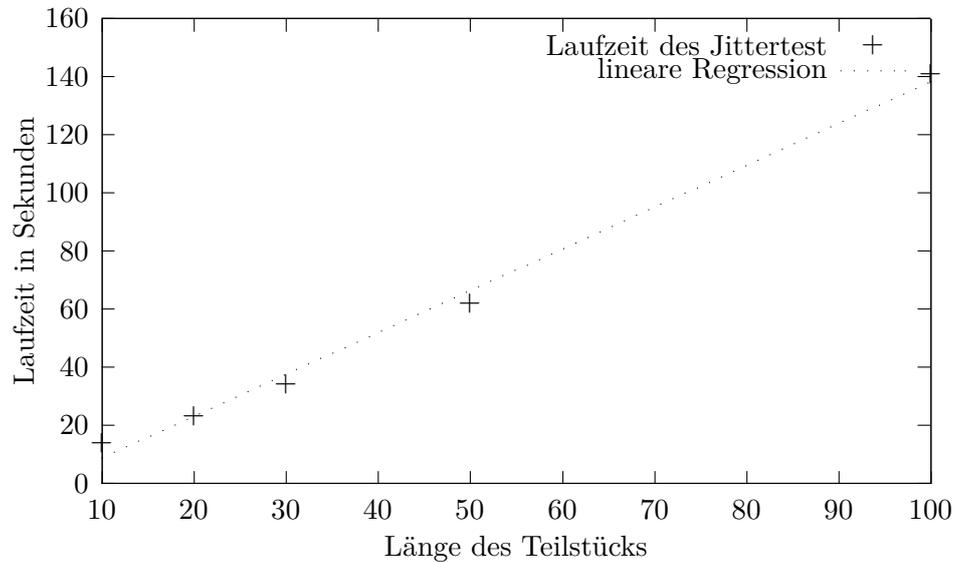
Beim Jittertest wurde aus den Daten ein Fluss ausgewählt und aus dem Polygonzug dieses Flusses ein Teilpolygonzug extrahiert. Um die Differenzen zwischen den Poly-

gonzügen aus der Datenbank und den später aus Bildern generierten Polygonzügen zu simulieren, wurde der extrahierte Teilpolygonzug verwackelt. Dazu wurden alle Punkte auf beiden Achsen unabhängig voneinander um (beschränkte) zufällige Werte verschoben. Für einen Satz von je 100 zufällig ausgewählten Abschnitten ergeben sich dabei die in Tabelle 4.1 dargestellten Erfolgsraten (in %) in Abhängigkeit von der Länge des Abschnitts (in Punkten) in den Zeilen und der maximalen Verwacklung (in  $10^{-5}$  Grad  $\approx$  Meter) in den Spalten.

**Tab. 4.1.:** Erfolgsraten des Jittertest in %

Länge des Teilstücks	Max. Verwacklung in $10^{-5}$ Grad					
	0	1	2	3	5	10
10	98	57	20	8	2	0
20	100	80	51	28	7	0
30	100	94	64	48	10	1
50	100	95	69	56	16	1
100	100	99	93	80	33	3

Abbildung 4.3 stellt die arithmetisch über dieselben Samples gemittelten Laufzeiten in Sekunden dar. Die Zeiten skalieren erwartungsgemäß nahezu linear mit der Länge des Testpolygonzuges und sind unabhängig von der Verwacklung.



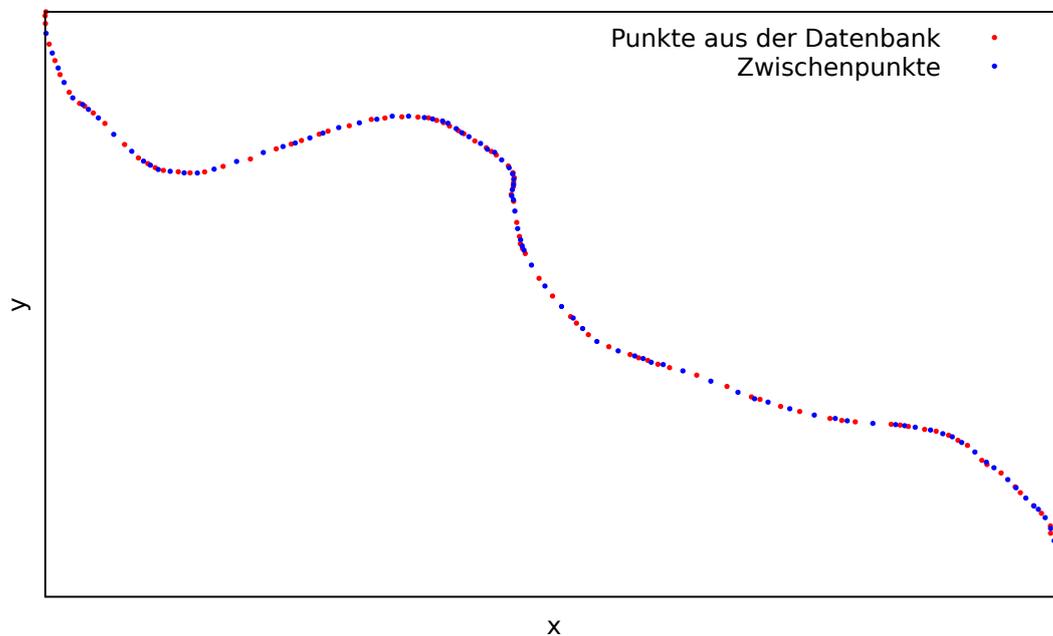
**Abb. 4.3.:** Laufzeiten des Jittertest in Sekunden

Es ergeben sich für die Suche folgende wichtige Parameter, die sowohl ausschlaggebend für dessen Erfolgchance, als auch für seine Laufzeit sind:

- Der genutzte Datensatz und ins besondere dessen Umfang
- Die Länge des Testpolygonzuges
- Die maximale Verwacklung des Testpolygonzuges

Dabei hängt die kompensierbare maximale Verwacklung erwartungsgemäß direkt von der Länge des Testpolygonzuges ab.

Beim Test zu alternativen Samplings wurden für Polygonzüge aus der Datenbank neue Punkte berechnet, die zwar auf derselben medialen Achse liegen, jedoch nicht zum ursprünglich Polygonzug gehören. Diese neuen Punkte bilden als Polygonzug ein alternatives Sampling derselben Samplerate zur selben medialen Achse mit entsprechend sehr ähnlichem Verlauf. Abbildung 4.4 zeigt einen solchen Polygonzug neben dem ursprünglichen, aus dem er berechnet wurde. Wie aus Tabelle 4.2 hervorgeht, ist selbst ohne Verwacklung keine zuverlässige Zuordnung mehr möglich.



**Abb. 4.4.:** Zwei verschiedene Polygonzüge entlang der selben medialen Achse

**Tab. 4.2.:** Erfolgsraten des Tests zu alternativen Samplings in %

Länge des Teilstücks	Max. Verwacklung in $10^{-5}$ Grad					
	0	1	2	3	5	10
100	2	3	2	3	1	0

#### 4.4.4. Simulationstest

Im vorherigen Abschnitt konnte gezeigt werden, unter welchen Bedingungen ein verwickeltes Teilstück eines Polygonzuges, der aus der Datenbank entnommen wurde, in der Datenbank wiedergefunden werden kann. Im nächsten Schritt sollte gezeigt werden, dass die Suche auch dann funktioniert, wenn der Teilpolygonzug händisch aus entsprechendem Bildmaterial gewonnen wurde. Da der Test zu alternativen Samplings fehlschlug, ist nicht davon auszugehen, dass der Simulationstest erfolgreich verläuft. Er wird deshalb nicht durchgeführt. Das Proof of Concept ist somit nicht erfolgreich.

#### 4.5. Analyse

Das Verfahren zur Bestimmung des RbFa beruht darauf, dass die Kanten zwischen aufeinander folgenden Punkten in den beiden zu vergleichenden Polygonzügen ähnliche Winkel zum Koordinatensystem aufweisen. Bestehen beide Polygonzüge aus unterschiedlichen Punkten auf derselben Funktion, haben die entstehenden Kanten stark voneinander abweichende Winkel, die durch den RbFa über die gesamte Länge aufintegriert werden. Somit ist das Ähnlichkeitsmaß erheblich von den gewählten Punkten abhängig.

Das Verfahren zur Bestimmung des RbFa liefert gute Ergebnisse bei der Suche, solange der Polygonzug des gesuchten Flussabschnittes mit ähnlichem Sampling erstellt wurde wie der zugehörige Fluss in der Datenbank. Weicht das Sampling durch starke Verwacklung, oder alternatives Sampeln entlang derselben Achse ab, liefert die Suche sehr schlechte Ergebnisse. Demnach ist der RbFa nicht invariant bezüglich alternativen Samplings. Da die zu suchenden Polygonzüge auf einem Bild generiert werden sollen, handelt es sich um Samplings, die im Allgemeinen große Abweichungen vom Sampling in der Datenbank aufweisen.

Somit ist der RbFa im Allgemeinen zum Vergleich von Polygonzügen, die Flüsse darstellen, nicht geeignet. Daraus ergibt sich die Forderung an winkelbasierte Ähnlichkeitsmaße, dass nur über alle Samplings stabile Winkel verwendet werden dürfen.

## 5. Realisierung mit Wendepunkten

Als logische Folge der oben beschriebenen Erkenntnisse erscheint eine Reduzierung der Kurve auf ihre Wendepunkte sinnvoll. Auch wenn die Position der Wendepunkte über verschiedene Samplings deutlich variieren kann, sind die Richtungen ihrer Tangenten dennoch stabil. Die Forderung an ein winkelbasiertes Ähnlichkeitsmaß ist somit erfüllt, wenn ausschließlich diese Richtungen verwendet werden. Auf dieser Basis erfolgt ein erneuter Realisierungsversuch, der nachfolgend beschrieben wird.

### 5.1. Datenaufbereitung

Die Darstellung eines Flussverlaufes enthält eine große Anzahl Wendepunkte. Hierbei kann in drei Kategorien unterschieden werden:

- Fehlerhafte Wendepunkte, die durch das Sampling ungewollt entstanden sind (Jitter)
- "Kleine" Wendepunkte, die für den Verlauf des Flusses nur geringe Relevanz haben
- "Große" Wendepunkte, die für den Verlauf des Flusses hohe Relevanz haben

Fehlerhafte Wendepunkte sind auf die Samplingrate und -qualität zurückzuführen. Daher verletzen sie die geforderte Samplinginvarianz und dürfen für die Flusserkennung nicht genutzt werden. Neben der Samplingrate haben auch Differenzen zwischen medialer Achse und Talweg des Flusses Einfluss auf die Entstehung von Wendepunkten. Diese "kleinen" Wendepunkte sind meist nur entweder in der Datenbank oder auf dem Bild vorhanden. Daher sind sie ebenfalls nicht Samplinginvariant und dürfen nicht verwendet werden. Die verbleibenden "großen" Wendepunkte charakterisieren den Verlauf des Flusses. Ziel der Datenaufbereitung ist es, eine neue Datenbank zu erstellen, die ausschließlich die Informationen der großen Wendepunkte enthält.

Um die Erkennung von fehlerhaften und kleinen Wendepunkten präventiv zu verhindern und die häufig innerhalb eines betrachteten Flussabschnittes inhomogene Samplingrate auszugleichen, wird vor der eigentlichen Berechnung der Wendepunkte ein Line-Smoothing-Algorithmus auf die Polygonzüge angewendet. Der verwendete "Sliding Averaging Algorithm" wurde von McMaster und Shea [MS92] entwickelt und im Rahmen dieser Arbeit parametrisiert implementiert.

Beim Smoothing wird über die Koordinaten der  $k$  Punkte vor und nach jedem Punkt arithmetisch gemittelt und der betrachtete Punkt in Abhängigkeit von der Gewichtung  $l$  gegen dieses Mittel verschoben. Diejenigen Punkte an Anfang und Ende des Polygonzuges, für die diese  $k$ -Umgebung nicht vollständig existiert, werden verworfen. Dies

führt dazu, dass Krümmungen aus bis zu etwa  $k$  Koordinaten begradigt werden. Somit ist  $k$  auch abhängig von der Samplerate des Polygonzuges und muss geeignet gewählt werden. Um sicherzustellen, dass der Informationsgehalt relevanter Krümmungen erhalten bleibt, wird das Smoothing mit geringer Gewichtung  $l$  durchgeführt. Damit unerwünschte Krümmungen dennoch eliminiert werden, wird das Smoothing rekursiv  $m$  mal durchgeführt. Folgende Parameter wurden für die nachfolgenden Tests verwendet:  $k = 2$ ,  $l = 0,5$  und  $m = 7$ .

Zur Bestimmung der Wendepunkte werden die durch das Smoothing erzeugten Polygonzüge genutzt. Hierbei liegt ein Wendepunkt vor, wenn sich das Vorzeichen der Krümmung in einem Punkt bezogen auf den vorherigen Punkt ändert. Der zugehörige Wendepunkt liegt dann zwischen den beiden Punkten und die Richtung seiner Tangente entspricht der Richtung der Kante zwischen den Punkten. Durch das durchgeführte Smoothing besitzen unerwünschte Wendepunkte flache Winkel bezüglich mindestens einem der benachbarten Wendepunkte. Aus diesem Grund werden Wendepunktpaare mit zu flachem Winkel ( $< 1,5$  Grad) zueinander verworfen und somit nur die "großen" Wendepunkte (Koordinaten und Winkel der Tangente) in die Datenbank aufgenommen.

## 5.2. Ähnlichkeitsmaß

Die Koordinaten der Wendepunkte dienen lediglich der Zuordnung des Ergebnisses einer Suche und die Winkel der Tangenten in den Wendepunkten sind somit alleiniger Informationsträger. Somit bestehen bereits drei Translationsinvarianzen durch die Begrenzung auf Winkel. Durch die weitere Einschränkung auf Winkeländerungen ist auch die Invarianz bezüglich Rotation gegeben.

Diese Winkeländerung zwischen zwei benachbarten Wendepunkten beschreibt die dazwischen liegende Kurve grob. Bildet man die Differenz zwischen den Winkeländerungen der beiden zugrundeliegenden Polygonzüge, erhält man ein Maß für die Ähnlichkeit dieser zwei Kurven. Summiert man dieses Maß über die Länge der betrachteten Abschnitte auf, erhält man ein Maß für die Ähnlichkeit der zugrunde liegenden Flussabschnitte, dass sich in  $O(n)$  bestimmen lässt, wobei  $n$  die Länge der beiden zu vergleichenden Abschnitte in Wendepunkten ist.

Das Verfahren ist im Allgemeinen nicht invariant bezüglich Verzerrung, sodass im Weiteren davon ausgegangen wird, dass die entsprechende Transformation bereits durchgeführt wurde. Auch hier könnte z.B. ein Accelerometer als Informationsquelle für die Transformation dienen.

## 5.3. Programmablauf

Bei dem verwendeten Ähnlichkeitsmaß gibt es für Teilstücke der Flüsse aus der Datenbank nur die Einschränkung, dass nach dem Startwendepunkt noch ausreichend viele Wendepunkte vorhanden sein müssen. Der Programmablauf sieht dabei grob wie folgt aus:

- Einlesen des Datensatz aus einer Datei in eine im RAM liegende Datenbank
- Zu findenden Polygonzug einlesen, glätten und Wendepunkte bestimmen
- Über alle Flüsse aus der Datenbank iterieren
  - Über alle zulässigen Startwendepunkte iterieren
    - \* Ähnlichkeitsmaß für die beiden Wendepunktfolgen bestimmen
    - \* Den gefundenen Flussausschnitt und sein Ähnlichkeitsmaß merken, falls dieses besser ist, als das bisher gemerkte
- Ausgeben der besten Zuordnung

Die Anzahl zu berücksichtigender Startwendepunkte bestimmt sich dabei zu:

$$\sum_{i=1}^F \max(0, N_i - n)$$

Wobei  $F$  die Anzahl der Flüsse in der Datenbank ist und  $N_i$  deren jeweilige Länge. Da auch für Flüsse, die zu kurz sind um einen in Frage kommenden Startwendepunkt zu haben, konstante Zeit brauchen ergibt sich für die Laufzeit einer Suche:

$$n \cdot \sum_{i=1}^F \max(0, N_i - n) + F$$

Da die Summe für den Fall  $F = 1$  (Datenbank besteht aus lediglich einem Fluss) maximal ist, lässt sich die Laufzeit durch  $n \cdot (N - n) + 1$  nach oben abschätzen. Dabei ist  $N = \sum_{i=1}^F N_i$  die Größe der Datenbank. Für den zu erwartenden Fall  $n \ll N$  ergibt sich somit eine asymptotische Laufzeit von  $\Theta(Nn)$ . Abbildung 5.1 bestätigt die lineare Abhängigkeit der Laufzeit von  $n$  bei konstantem  $N$  experimentell.

## 5.4. Benchmark

Wie schon in der Testmethodik (Kapitel 3) beschrieben, muss die Vergleichsystematik zum Beweis der Verwendbarkeit die nachfolgenden Tests bestehen. Zu beachten ist hierbei, dass bei dieser Methodik ein paarweises Matching gesucht wird, sodass ein Ergebnis genau dann korrekt ist, wenn der richtige Fluss mit dem richtigen Anfangspunkt gefunden wird.

### 5.4.1. Kalibrierung

Der Integritätstest liefert erwartungsgemäß das Ähnlichkeitsmaß 0. Für das verwendete Ähnlichkeitsmaß werden ausschließlich die Winkel der Wendepunkt tangente verwendet. Die Koordinaten werden dabei somit nicht verwendet, sodass die Tests zu Translations-

und Skalierungsinvarianz mit dem Integritätstest identisch sind und somit ebenfalls erwartungsgemäß das Ähnlichkeitsmaß 0 bestätigen. Die drei Invarianzen bezüglich Translation sind somit trivial gegeben.

Da sich das Ähnlichkeitsmaß auf Winkeländerungen und nicht auf absolute Winkel bezieht, ist die Rotationsinvarianz auf Grund des Verfahrens bereits gegeben. Ein Test ist somit nicht erforderlich. Wie beschrieben, ist eine Verzerrungsinvarianz nicht gefordert und wird entsprechend nicht getestet.

#### 5.4.2. Suchalgorithmus

Der Test zur Richtungsinvarianz bestätigt, dass das Ähnlichkeitsmaß für Wendepunkte und ihre Inverse im Allgemeinen nicht 0 ist. Sofern die Fließrichtung des Flusses nicht bekannt ist, muss die Suche in beiden Richtungen durchgeführt werden. Dabei ist zu beachten, dass die Tangentenwinkel Richtungsinformation enthalten und daher im zweiten Durchlauf sowohl die Winkel, als auch ihre Reihenfolge invertiert werden müssen. Im Folgenden wird davon ausgegangen, dass die Fließrichtung bekannt ist.

Das Ähnlichkeitsmaß sieht eine exakte paarweise Zuordnung der betrachteten Wendepunkte vor. Daher ist es nur auf Abschnitten gleicher Länge definiert. Dies wird durch den Suchalgorithmus kompensiert, indem jeder mögliche Teilabschnitt eines Flusses getestet wird. Entsprechend ist das Ähnlichkeitsmaß für falsche Teilstücke im Allgemeinen nicht 0, sodass der Test zu speziellen partiellen Zuordnungen entfallen kann.

Der Test zu allgemeinen partiellen Zuordnungen kann hier als mehrfache Ausführung des Tests zu speziellen partiellen Zuordnungen angesehen werden. Getestet wurde mit jeweils 100 zufälligen Teilabschnitten unterschiedlicher Länge. Wie in der ersten Spalte der Tabelle 5.1 dargestellt, können bereits mit Teilstücken von drei Wendepunkten Länge sehr gute Ergebnisse erzielt werden. Somit wurde der Suchalgorithmus positiv getestet.

#### 5.4.3. Samplingtests

Aufgrund der Datenaufbereitung (Abschnitt 5.1) liegt ein sehr spezielles Sampling mit fester Samplingrate (ein Wendepunkt pro Richtungsänderung) vor. Somit sind Tests zu verschiedenen Sampleraten und zu alternativen Samplings hier nicht anwendbar. Die zu kompensierende Unschärfe der Merkmale beschränkt sich daher ausschließlich auf die Winkel der Wendetangenten. Die Toleranz des Verfahrens wurde im Jittertest durch zufällige Verfälschung der einzelnen Winkel durchgeführt. In Tabelle 5.1 ist die Erkennungsrate über ein Sample von je 100 zufälligen Teilstücken in Abhängigkeit von der Länge des Teilstücks und der maximalen Verwacklung in Grad dargestellt. Erwartungsgemäß steigt die kompensierbare Verwacklung mit der Länge der verwendeten Teilstücke.

Abbildung 5.1 stellt die arithmetisch über jeweils 600 Samples gemittelten Laufzeiten in Millisekunden dar. Daraus ergibt sich erwartungsgemäß, dass der Einfluss der Länge des Teilstücks auf die Laufzeit leicht geringer als linear ist. Dies ist darauf zurückzuführen, dass mit steigender Länge des Teilstücks die Anzahl der Schleifeniterationen sinkt.

**Tab. 5.1.:** Erfolgsraten in %

Länge des Teilstücks	Maximale Verwacklung in Grad					
	0	2	5	10	15	20
3	100	2	1	0	0	0
5	100	75	28	8	3	0
10	100	100	95	75	48	28
15	100	100	99	97	88	47
20	100	100	100	100	98	79

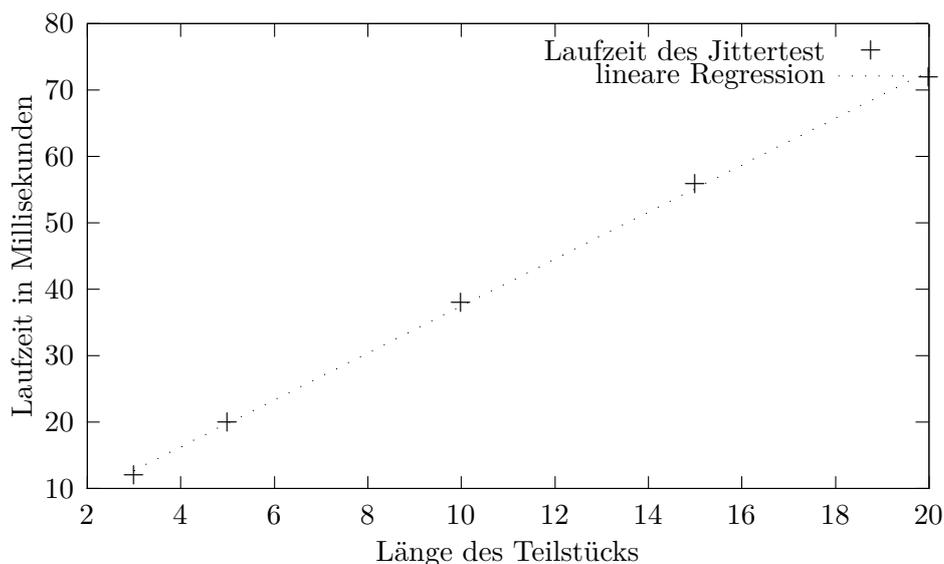
**Abb. 5.1.:** Laufzeiten des Jittertest in Millisekunden

Tabelle 5.2 zeigt für die selben Samples die Anteile unter denen das korrekte Ergebnis mindestens das drittbeste Ergebnis der Suche war. Daraus lässt sich ablesen, dass die Erkennungsrate durch externe Einschränkungen, wie zum Beispiel eine sehr grobe Vorgabe des Breitengrades, teilweise deutlich verbessert werden könnte. Bei hohem Verhältnis *Verwacklung/Länge* (z.B. 20/3) werden die Ergebnisse allerdings beliebig schlecht und dieser Ansatz führt zu keiner signifikanten Verbesserung.

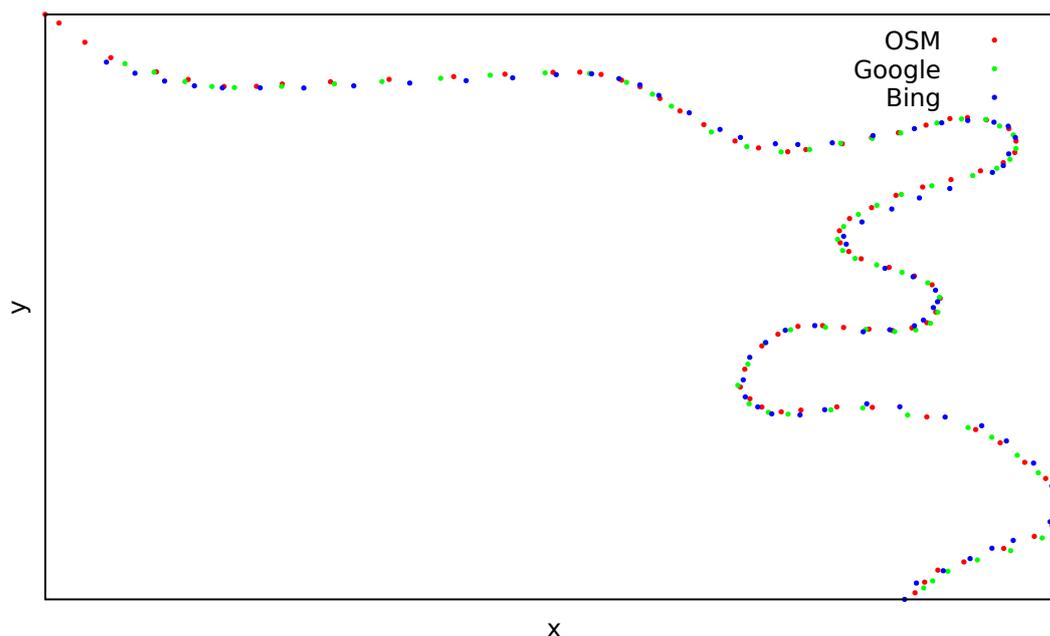
#### 5.4.4. Simulationstest

Für den Simulationstest wird der Polygonzug, der den Fluss auf einem Bild beschreibt, manuell durch einen Menschen festgelegt. Aus lizenztechnischen Gründen wurden die verwendeten Bilder durch OpenStreetMaps generiert. Um sicherzustellen, dass keine abhängigen Ergebnisse durch die Verwendung der selben Datenquelle entstanden sind, wurden auch Bilder von unabhängigen (lizenzpflichtigen) Quellen zur Verifizierung genutzt.

**Tab. 5.2.:** Anteil in den Top drei Ergebnissen in %

Länge des Teilstücks	Maximale Verwacklung in Grad					
	0	2	5	10	15	20
3	100	9	2	0	0	0
5	100	90	44	14	5	2
10	100	100	100	83	64	43
15	100	100	100	99	93	63
20	100	100	100	100	99	93

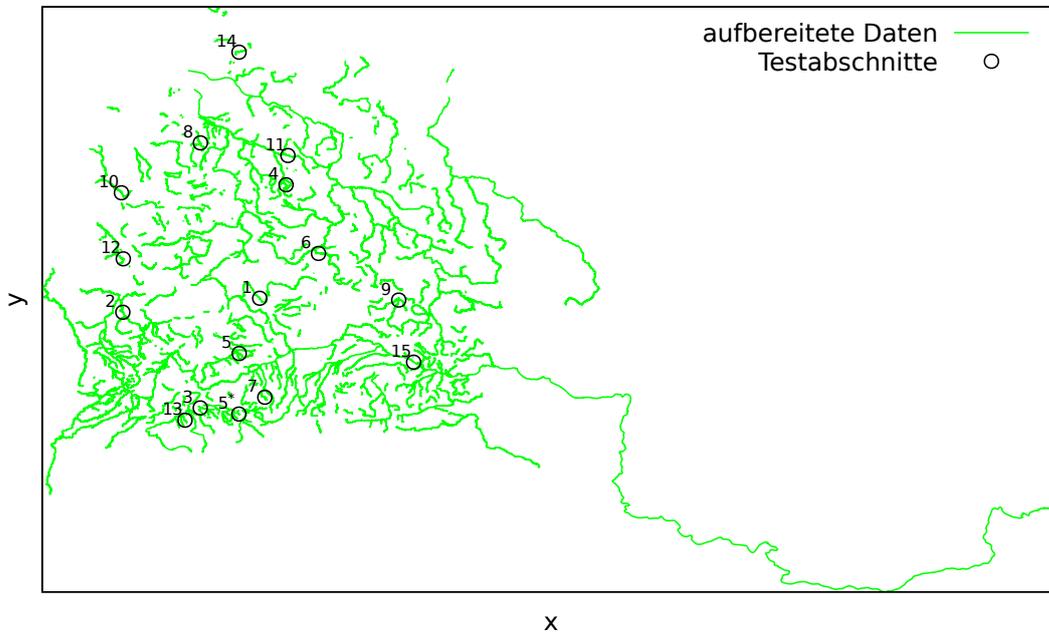
Abbildung 5.2 zeigt die, durch manuelles sampeln auf Bildern verschiedener Quellen, leicht abweichenden Darstellungen des selben Flussabschnittes. Die Polygonzüge werden so über die Bilder gezeichnet, dass die verwendeten Kanten grob den medialen Achsen entsprechen. Die (Pixel-)Koordinaten der dazu verwendeten Punkte definieren den Polygonzug und sind die Informationsträger, die für die Verarbeitung genutzt werden. In einem automatischen Flusserkennungssystem wird diese Aufgabe von einer Software übernommen.



**Abb. 5.2.:** manuell gesampelte Polygonzüge unterschiedlicher Quellen

Der manuell erzeugte Polygonzug wird eingelesen und im Folgenden automatisch verarbeitet. Aus dem eingelesenen Polygonzug werden mit den Verfahren aus der Datenaufbereitung (Abschnitt 5.1) die Wendepunkte des betrachteten Abschnitts bestimmt. Mit diesen Wendepunkten wird der Suchalgorithmus aufgerufen. Als Ergebnis werden die

Koordinaten des ersten Wendepunktes des ermittelten Flussabschnittes ausgegeben. Die Überprüfung des Ergebnisses wird manuell durchgeführt. Abbildung 5.3 zeigt die geographische Einordnung der größtenteils zufällig ausgewählten und im Folgenden verwendeten Flussabschnitte. Die Namen und nächstgelegenen Ort der getesteten Flussabschnitte sind in Tabelle 5.3 aufgelistet.



**Abb. 5.3.:** getestete Flussabschnitte

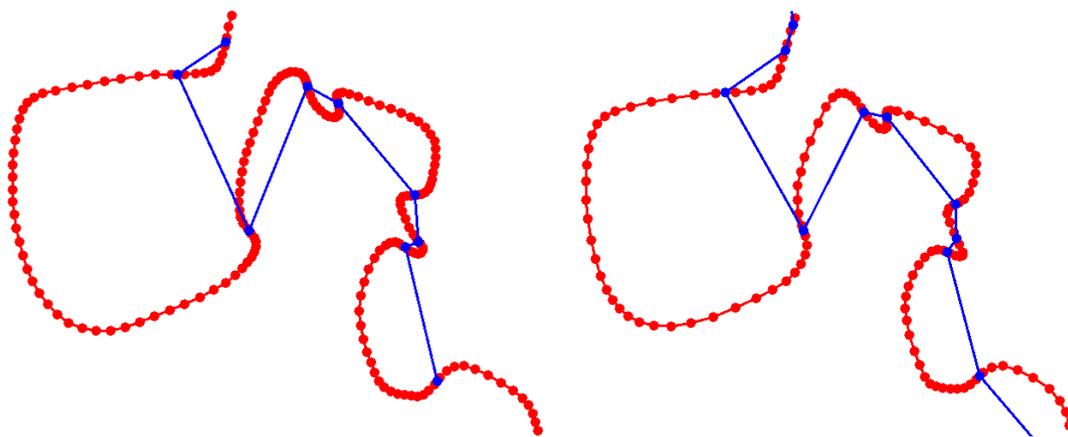
Im Simulationstest wurden insgesamt 15 verschiedene Flussabschnitte von Bildern gesucht. Mit 12 korrekten Erkennungen wurde eine Erfolgsrate von 80% erreicht. Nicht korrekt erkannt wurden die Abschnitte 1, 5 und 11. Bei den im Folgenden dargestellten Abbildungen wird links der gesuchte Flussabschnitt dargestellt und rechts das korrekte Ergebnis aus der Datenbank. Die roten Punkte und Linien markieren dabei jeweils den aufbereiteten Polygonzug des Flussverlaufs. Die blauen Punkte markieren die Wendepunkte, deren Verbindungslinien hier nicht relevant sind. Nachfolgend werden einige der Testergebnisse, insbesondere alle mit negativem Ergebnis, vorgestellt. Die vollständige Zusammenstellung aller getesteten Flussabschnitte befindet sich im Anhang.

Abbildung 5.4 zeigt den Fluss Úhlava in Plzeň, der mit seinem signifikanten Verlauf für das Verständnis der Vergleichssystematik gut geeignet ist und korrekt erkannt wurde. Deutlich zu erkennen ist die erhebliche Reduktion des genutzten Informationsumfangs durch die Beschränkung auf Wendepunkte. Ebenfalls deutlich zu erkennen ist die relativ geringe Empfindlichkeit des Verfahrens gegenüber der Samplingrate. Betrachtet man jedoch den Wendepunkt in der Mitte (im linken Graphen der Dritte von oben), so lässt sich ein deutlicher Unterschied zwischen den dargestellten Winkeln erkennen. Beim Smoothing bewirkt eine bezüglich  $k$  geringe Anzahl an Punkten in der Umgebung

**Tab. 5.3.:** geteste Flussabschnitte

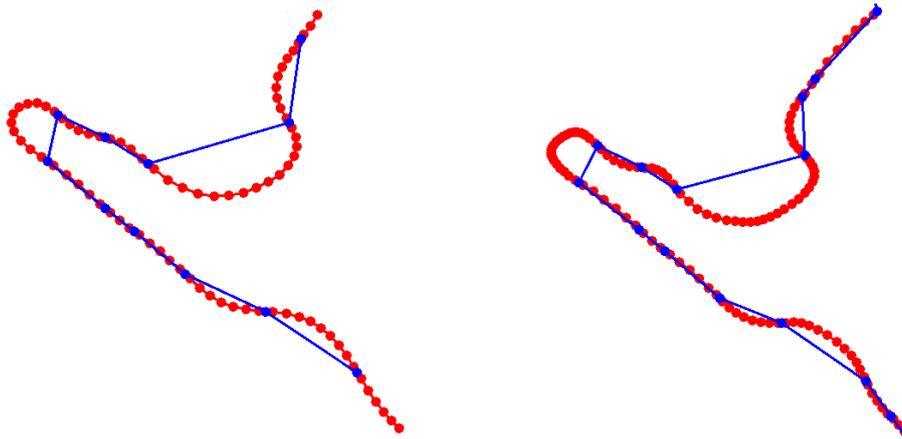
Nummer	Name	Ort
1	Main	Würzburg
2	Saar	Mettlach
3	Glatt	Schachen
4	Ecker	Abbenrode
5	Lindach	Kirchheim unter Teck
5*	Goldach	Trogen
6	Saale	Saalfeld
7	Eschach	Friesenhofen
8	Hunte	Barnstorf
9	Úhlava	Plzeň
10	Issel	Loikum
11	Aller	Osloss
12	Urft	Urft
13	Wyna	Zetzwil
14	Sorge	Königsberg
15	Donau	Schlogen

des Wendepunktes ein Kippen der Tangente. Da der Informationsgehalt eines Punktes abhängig von der lokalen Krümmung ist, werden Kurven üblicherweise höher gesampelt als Geraden, auf denen die Wendepunkte liegen. Daher ist dieses Problem im genutzten Smoothing-Algorithmus bei konstantem  $k$  nicht lösbar und führt zu einer Verschlechterung der Datenbasis.

**Abb. 5.4.:** Testabschnitt 9: Úhlava in Plzeň

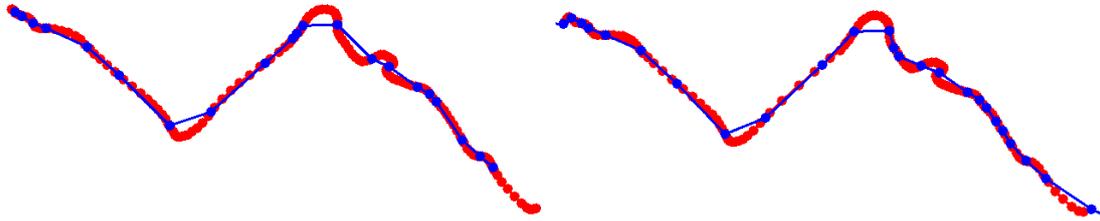
Weiterhin kann gezeigt werden, dass Polygonzüge von Bildern an den Rändern weniger Wendepunkte enthalten können als in der Datenbank (im Bild oben). Das liegt daran, dass vom Rand des Polygonzugs bis zu einem solchen Wendepunkt zu wenig Krümmung vorhanden sein kann, um diesen noch eindeutig als solchen zu identifizieren. Für die Erkennung ist dieser Randeffekt ohne Bedeutung, da das Matching der verbleibenden Wendepunkt nicht beeinträchtigt wird.

Abbildung 5.5 zeigt die ebenfalls korrekt erkannte Saar in der Nähe von Mettlach (Saarschleife). In diesem Flussabschnitt befinden sich mehrere Wendepunkte auf einem optisch nahezu linearen Teilabschnitt. Deshalb kann mit diesem Beispiel gezeigt werden, dass auch die Tangenten von "großen" Wendepunkten kleiner Kurven durch die Datenaufbereitung flacher werden. Die Winkeldifferenzen dieser Tangenten fallen hierbei gerade noch nicht unter das Ausschlusskriterium für "kleine" und fehlerhafte Wendepunkte. Trifft dieser Umstand auf einen Flussabschnitt zu, der gleichzeitig ungenau gesampelt wurde, kann das dazu führen, dass "große" Wendepunkte durch die Datenaufbereitung fälschlicherweise aussortiert werden. Dies ist zum Beispiel am oberen Ende des betrachteten Flussabschnitts aufgetreten. Weil es sich um einen Randbereich handelt, führte dieser Fehler jedoch nicht zu einer Beeinträchtigung des Matchings.



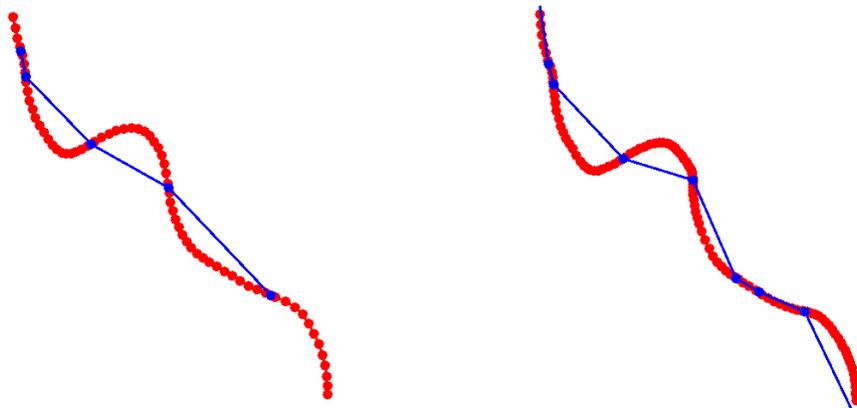
**Abb. 5.5.:** Testabschnitt 2: Saarschleife bei Mettlach

Abbildung 5.6 zeigt die Aller bei Osloss in der Nähe von Wolfsburg. Auf diesem Flussabschnitt liegen die durch die Datenaufbereitung auf dem Bild fälschlicherweise aussortierten Wendepunkte nicht am Rand. Die so entstandene Diskrepanz im Matching der Wendepunkte führt dazu, dass dieser Flussabschnitt nicht korrekt erkannt wird. Durch ein genaueres Sampling auf dem Bild kann die Auswirkung dieses Effektes in vielen Fällen verhindert werden. Grundsätzlich kann jedoch nicht immer verhindert werden, dass einzelne Wendepunktpaare in der Datenbank oder auf dem Bild als "groß" identifiziert werden und auf dem entsprechenden Gegenstück als "klein". Dies liegt daran, dass sich die zugrunde liegenden Achsen, auf denen gesampelt wird, leicht unterscheiden können.



**Abb. 5.6.:** Testabschnitt 11: Aller bei Osloss in der Nähe von Wolfsburg

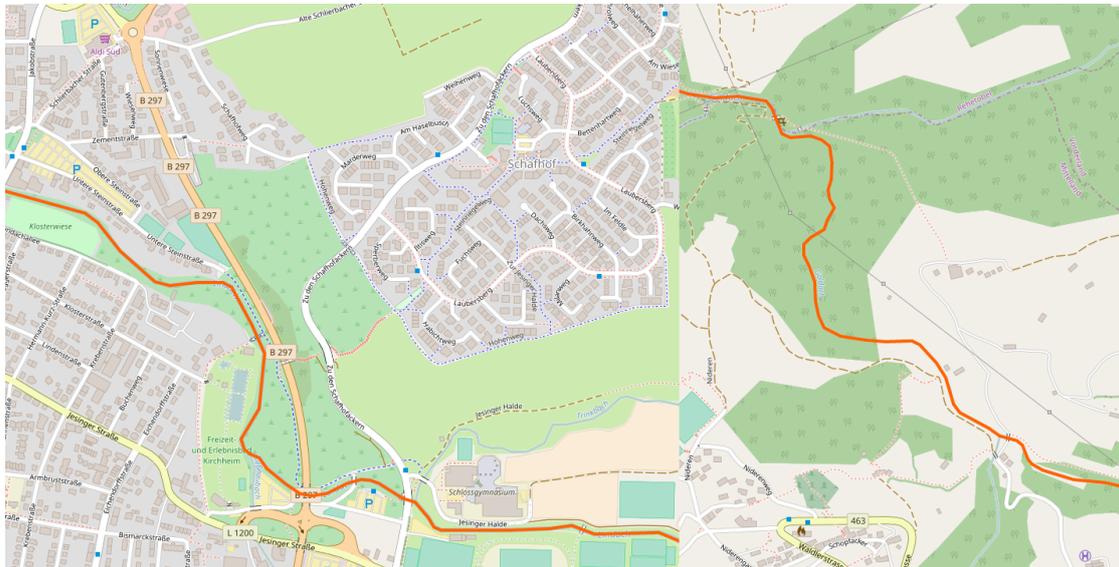
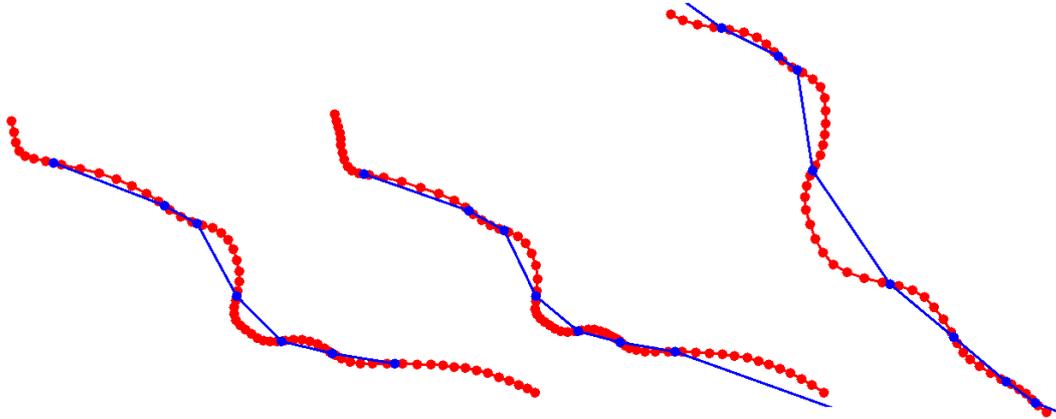
Am Beispiel des Maines in Würzburg (Abbildung 5.7) kann dieses Verhalten für die Kurve zwischen dem dritten und vierten Wendepunkt von unten im Datenbank-Ausschnitt (rechts) beobachtet werden. Der Winkel dieser Kurve liegt sehr nahe oberhalb der Ausschlussgrenze. Daher ist die Wahrscheinlichkeit, dass der selbe Winkel in einem anderen Sampling zufällig unterhalb der Grenze liegt, recht hoch. Aus diesem Grund wurde dieser Flussabschnitt nicht erkannt.



**Abb. 5.7.:** Testabschnitt 1: Main in Würzburg

Das Beispiel der Lindach in Kirchheim unter Teck (Abbildung 5.8) zeigt auf den ersten Blick identische Wendepunkte in beiden Graphen. Dennoch wurde dieser Flussabschnitt nicht erkannt. Erkannt wurde stattdessen die rechts dargestellte Goldach bei Trogen, die ebenfalls auf den ersten Blick identische Wendepunkte aufweist. Eine genauere Analyse zeigt, dass sich die beiden Flussabschnitte in der Datenbank extrem ähnlich sind. Ein

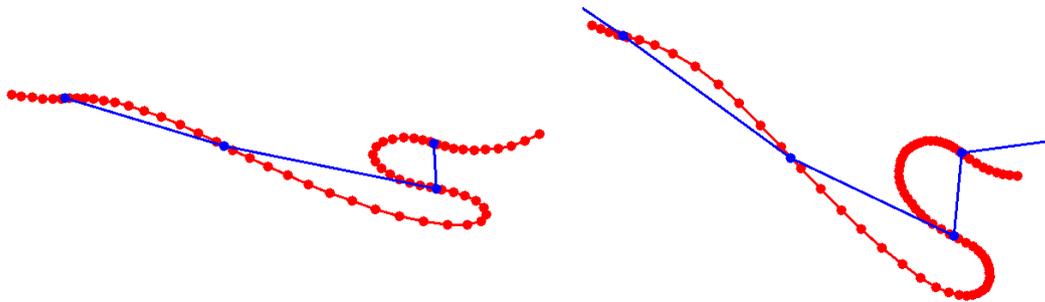
Vergleich der beiden Flussabschnitte ohne Datenaufbereitung bestätigt den ähnlichen Verlauf. Die verbleibenden Unterschiede wurden durch das Smoothing weitgehend beseitigt, sodass die beiden Flussabschnitte unter dieser Vergleichssystematik nahezu identisch sind. Eine korrekte Erkennung kann daher nicht erwartet werden.



**Abb. 5.8.:** Testabschnitt 5/5\*: Lindach in Kirchheim unter Teck und Goldach bei Trogen

Das in Abbildung 5.9 dargestellte Luftbild zeigt die Donau bei Schloggen und wurde korrekt erkannt. Das Bild wurde unter einem Winkel von schätzungsweise 30 Grad aufgenommen und ist somit deutlich verzerrt. Die hier vorgestellte Vergleichssystematik ist zwar nicht invariant gegen Verzerrungen, kann diese aber bis zu einem gewissen Grad kompensieren, sofern eine ausreichende Anzahl an Wendepunkten im betrachteten Flussabschnitt vorhanden ist.

Obwohl einige Flussabschnitte nicht erkannt wurden, kann der Simulationstest im Sinne eines Proof of Concept als bestanden gewertet werden. Die auf Wendepunkten



**Abb. 5.9.:** Testabschnitt 15: Donau bei Schloggen (Quelle: By Aisano (Own work) [GFDL (<http://www.gnu.org/copyleft/fdl.html>) or CC BY-SA 4.0-3.0-2.5-2.0-1.0 (<http://creativecommons.org/licenses/by-sa/4.0-3.0-2.5-2.0-1.0>)], via Wikimedia Commons)

basierte Vergleichssystematik ist somit prinzipiell zum Beschreiben und Identifizieren von Flussabschnitten geeignet.

## 6. Fazit

Im Rahmen dieser Arbeit konnten wichtige Erkenntnisse zur Positionsbestimmung auf der Basis von Flussverläufen gesammelt werden. Eine Ermittlung der Position ist grundsätzlich möglich, hängt jedoch von mehreren Faktoren ab. Ein wesentlicher Faktor ist die Qualität der verwendeten Vergleichsdaten, welche im vorliegenden Fall zwar ausreichend war, aber noch deutliches Verbesserungspotential bietet. Die Wahl geeigneter Merkmale zur Beschreibung eines Flusslaufes ist zusammen mit dem Verfahren zur Bestimmung eines Ähnlichkeitsmaßes der wichtigste Faktor für den Erfolg einer solchen Positionsbestimmung. In Kapitel 4 konnte gezeigt werden, dass Polygonzüge in Verbindung mit dem vergleichsweise aufwendigen Ähnlichkeitsmaß des richtungsbasierten Fréchetabstands auf Grund der Samplingvarianz nicht geeignet sind.

In einem anderen Ansatz wurde in Kapitel 5 beschrieben, dass die Wendepunkte eines Flusslaufes erfolgreich als Merkmale für die Positionsbestimmung eingesetzt werden können. Durch die erhebliche, verlustbehaftete Komprimierung der Informationen in die Wendepunkte kann ein vergleichsweise einfaches Ähnlichkeitsmaß definiert werden. Die verringerte Anzahl an Merkmalen verbessert dabei die Laufzeit der Suche in erheblichem Ausmaß. Der dabei erreichbare Grad an Samplinginvarianz hängt maßgeblich von der Qualität der verwendeten Datenbasis ab. Daher ist die Datenaufbereitung (Abschnitt 5.1), bei welcher die Polygonzüge geglättet und Wendepunkte auf ihnen bestimmt werden, für den Erfolg der Vergleichssystematik von entscheidender Bedeutung. Mit den Simulationstests (Abschnitt 5.4.4) konnte gezeigt werden, dass schon mit einer simplen händischen Optimierung der Parameter zur Datenaufbereitung eine Erkennungsrate von 80% erreicht werden kann. Es ist zu erwarten, dass diese Erkennungsrate mit geeigneten Optimierungsverfahren noch deutlich gesteigert werden kann. Auch wenn nicht explizit gefordert, ist dieses Verfahren wahrscheinlich recht robust gegenüber Verzerrungen durch eine gekippte Kamera.

Es konnte gezeigt werden, dass Wendepunkte, die auf der Basis von Polygonzügen bestimmt wurden, ausreichend Informationsgehalt zur Beschreibung von Flussläufen aufweisen. Daraus ergibt sich, dass auch in den Polygonzügen ausreichende Informationen enthalten sein müssen. Mit einem geeigneten Ähnlichkeitsmaß muss es daher möglich sein, mit Polygonzügen eine Erkennungsrate zu erreichen, die mindestens so hoch ist, wie die hier auf der Basis von Wendepunkten erreichte.

## 7. Ausblick

Nachdem in dieser Arbeit für die Positionsbestimmung mit Flüssen ein Proof of Concept erfolgreich abgeschlossen wurde, werden im Folgenden mögliche Optimierungen und Anwendungen aufgezeigt.

Es hat sich gezeigt, dass die Erkennungsrate in erheblichem Maße vom Smoothing abhängig ist, welches wiederum vom Sampling des Fluss(-abschnitts) beeinflusst wird. Die Smoothing-Parameter müssen daher entweder optimal gewählt, oder dynamisch an das jeweilige Sampling angepasst werden. Bei bekannter Sampling-Range der verwendeten Daten können z.B. durch ein lineares Programm oder ein neuronales Netz die optimalen Parameter bestimmt werden. Ist die Sampling-Range unbekannt oder sehr weit gestreut, können mit statischen Parametern keine beliebig guten Ergebnisse mehr erzielt werden. In einem solchen Fall können Smoothing-Parameter, die dynamisch an das Sampling des jeweiligen Polygonzugs angepasst werden, für die Optimierung der Ergebnisse verwendet werden. So kann beispielsweise der Parameter  $k$ , welcher wesentlichen Einfluss auf den Ausschluss unerwünschter Wendepunkte hat, in Abhängigkeit von der durchschnittlichen Krümmungsrates des Polygonzuges bestimmt werden. Der Einfluss der Samplerate auf die Datenaufbereitung kann so je nach Sampling gemindert oder komplett beseitigt werden. Eine weitere Möglichkeit ist,  $k$  in Abhängigkeit von der lokale Krümmung zu bestimmen und somit zusätzlich das Kippen der Wendetangenten zu minimieren.

In der vorliegenden Form erfordert die Bestimmung des Ähnlichkeitsmaßes ein perfektes Matching der Wendepunkte. Befindet sich die Winkeldifferenz eines Wendepunktpaares nahe an der Ausschlussgrenze kann, wie bereits anhand der Abbildung 5.7 beschrieben, dieses perfekte Matching beschädigt werden. Wird die Suche nach kostenminimalen Matchings auf Matchings erweitert, die sich lediglich durch wohl definierte Lücken von perfekten unterscheiden, könnten die beschriebenen Flussabschnitte dennoch korrekt erkannt werden.

Wie bereits im Fazit (Kapitel 6) beschrieben scheint die verwendete Vergleichssystematik eine relative hohe Toleranz bezüglich des Winkels aufzuweisen, unter dem das zugrunde liegende Bild aufgenommen wurde. Weitere Tests zum Bestimmen der Grenzen dieser Toleranz erscheinen, auch unter dem Gesichtspunkt eventuell keine Transformation durchführen zu müssen, sinnvoll.

Die hier erreichten Laufzeiten sind für den verwendeten Datensatz ausreichend. Für eine Verwendung des Verfahrens in Flugzeugen muss der Datensatz jedoch erheblich erweitert werden, wodurch sich die Laufzeit deutlich verschlechtern wird. Eine mögliche Kompensation bietet die Realisierung der Datenbank als generalisierter Suffixbaum (GST), indem die Knoten Wendepunkten und die Kanten den Winkeldifferenzen entsprechen. Zu beachten ist, dass die Winkeldifferenzen als Zeichensatz diskretisiert werden müssen, damit nach Galil und Giancarlo [GG88] eine Volltextsuche anwendbar ist. Die Laufzeit

einer Suche auf dieser Datenstruktur ist lediglich von der Anzahl der Wendepunkte, die für eine eindeutige Zuordnung notwendig sind, abhängig und somit insbesondere weder vom Umfang des Datensatzes noch von der Länge der Eingabe linear abhängig. Die erreichbare asymptotische Laufzeit liegt dabei in  $O(\log_Z N)$ , wobei  $Z$  die Größe des verwendeten Zeichensatzes ist. Der Performance-Gewinn durch die Verwendung des GST ist daher insbesondere bei großen Datensätzen, wie z.B. mit allen Flüssen der Erde, erheblich. Zusätzlich bietet dieser Ansatz eine Lösung für den im Abschnitt 4.1 geforderten eindeutigen Flussverlauf, da multiple Verläufe desselben Flusses im GST darstellbar sind.

Die in dieser Arbeit als gegeben angenommene automatische Erkennung der Flussverläufe auf einem Bild ist derzeit Gegenstand einer weiteren Arbeit. In Kombination mit dieser Arbeit ist dann eine automatische autonome Positionsbestimmung mit Hilfe von Luftbildern möglich.

Eine Schwäche des hier beschriebenen Verfahrens wird insbesondere dann deutlich, wenn ein Bild aus niedriger Flughöhe aufgenommen wird, da die Wahrscheinlichkeit, einen Flussabschnitt ausreichender Länge abgebildet zu haben, relativ gering ist. Daher sind weitere Informationsträger erforderlich, um eine flächendeckende Positionsbestimmung zu ermöglichen. Denkbar wäre die Verwendung weiterer Verkehrswege, wie z.B. das Schienen- oder Straßennetz. Wegen seiner großen Verbreitung und der leichten Identifizierung auf Bildern bietet sich hier das Straßennetz an. Um auf der Basis eines, auf einem Bild erkannten, Teilnetzes eine Positionsbestimmung durchzuführen, können zwar ähnliche Konzepte verwendet werden, aber die konkrete Vorgehensweise muss speziell hierfür entwickelt werden. Eine geeignete Vergleichssystematik wurde bereits von Funke und Storandt [FS11] in ihrer Arbeit zu Pfad-Formen entwickelt. Ziel dieser Arbeit war es, eine autonome Lokalisierung eines Fahrzeuges auf der Grundlage der Form des bereits zurückgelegten Weges zu realisieren. Gelingt es, die autonome Positionsbestimmung mithilfe von Flüssen und Straßen zu kombinieren, kann weltweit bereits in geringer Flughöhe eine gute Erkennungsrate für einen großen Teil des besiedelten Festlandes erzielt werden.

# Literaturverzeichnis

- [Blu67] Harry Blum: A Transformation for Extracting New Descriptors of Shape. In: W. Wathen-Dunn (Herausgeber): *Proc. Models for the Perception of Speech and Visual Form*, Seiten 362–380, 1967.
- [dBC11] Mark de Berg und Atlas F. Cook: Go with the Flow: The Direction-Based Fréchet Distance of Polygonal Curves. In: *Proc. 1st Int. Conf. on Theory and Practice of Algorithms in (Computer) Systems (TAPAS'11)*, Seiten 81–91, 2011. [http://dx.doi.org/10.1007/978-3-642-19754-3\\_10](http://dx.doi.org/10.1007/978-3-642-19754-3_10).
- [FH12] Maurice R. Fréchet und Horace B. Heywood: *L'Équation de Fredholm et ses applications à la physique mathématique*. A. Hermann, Paris, 1912.
- [FS11] Stefan Funke und Sabine Storandt: Path shapes: An alternative method for map matching and fully autonomous self-localization. In: *Proc. 19th ACM SIGSPATIAL Int. Symp. on Advances in Geographic Information Systems (ACM-GIS'11)*, Seiten 319–328, 2011. <http://doi.acm.org/10.1145/2093973.2094016>.
- [GG88] Zvi Galil und Raffaele Giancarlo: Data structures and algorithms for approximate string matching. *Journal of Complexity*, 4(1):33–72, 1988. <http://www.sciencedirect.com/science/article/pii/0885064X88900088>.
- [Gro99] Misha Gromov: *Metric Structures for Riemannian and Non-Riemannian Spaces*. Birkhäuser, Bosten, 1999.
- [Hau14] Felix Hausdorff: *Grundzüge der Mengenlehre*. Veit & Co., Leipzig, 1914.
- [JFZ<sup>+</sup>14] Hao Jiang, Min Feng, Yunqiang Zhu, Ning Lu, Jianxi Huang und Tong Xiao: An Automated Method for Extracting Rivers and Lakes from Landsat Imagery. *Remote Sensing*, 6(6):5067–5089, 2014. <http://dx.doi.org/10.3390/rs6065067>.
- [MS92] Robert Brainerd McMaster und K. Stuart Shea: *Generalization in Digital Cartography*. Association of American Geographers, Washington, D.C., 1992.
- [TWC<sup>+</sup>12] Ziheng Tian, Chengdong Wu, Dongyue Chen, Xiaosheng Yu und Li Wang: A Novel Method of River Detection for High Resolution Remote Sensing Image Based on Corner Feature and SVM. In: *Proc. 9th Int. Symp. on Neural Networks (ISNN'12)*, Seiten 266–273, 2012. [http://dx.doi.org/10.1007/978-3-642-31362-2\\_30](http://dx.doi.org/10.1007/978-3-642-31362-2_30).

# Erklärung

Hiermit versichere ich die vorliegende Abschlussarbeit selbstständig verfasst zu haben, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt zu haben, und die Arbeit bisher oder gleichzeitig keiner anderen Prüfungsbehörde unter Erlangung eines akademischen Grades vorgelegt zu haben.

Würzburg, den 12. Juli 2017

.....  
Felix Klesen

## A. Anhang: Testdaten

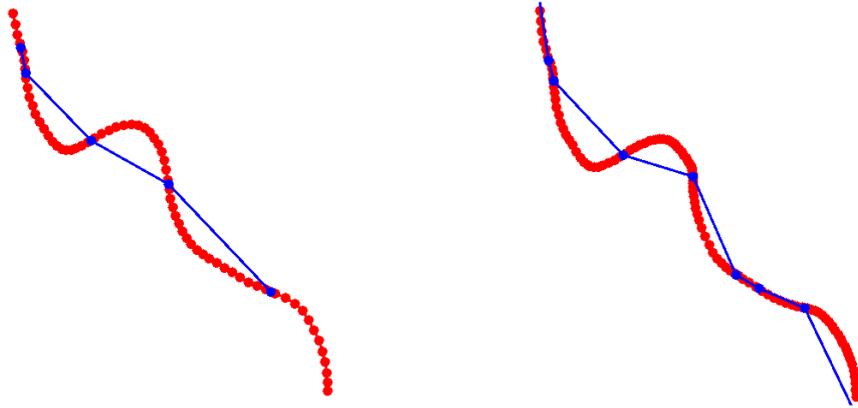


Abb. A.1.: Testabschnitt 1: Main in Würzburg

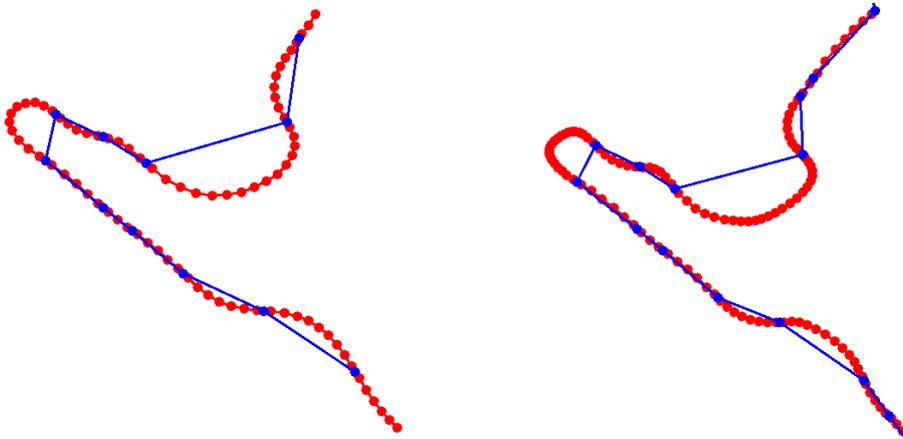


Abb. A.2.: Testabschnitt 2: Saar bei Mettlach

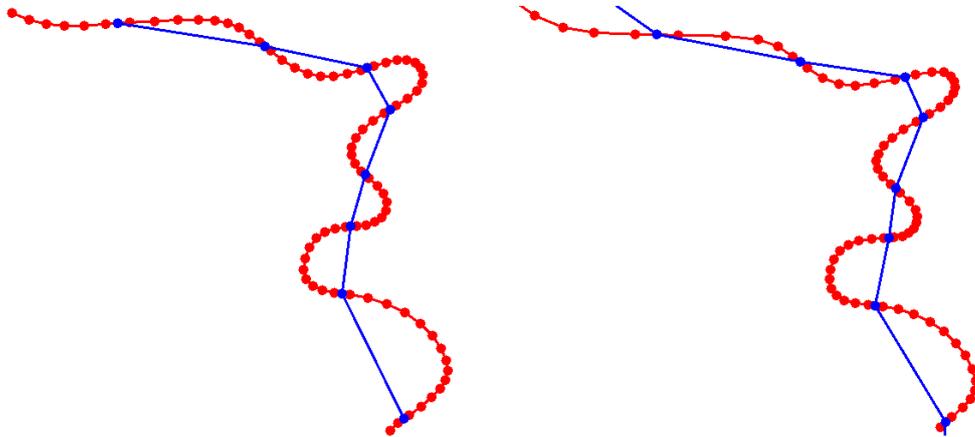


Abb. A.3.: Testabschnitt 3: Glatt bei Schachen

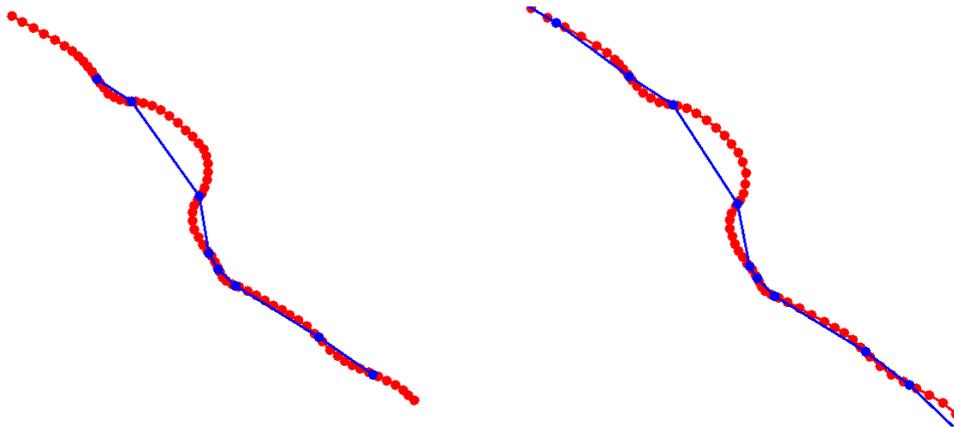


Abb. A.4.: Testabschnitt 4: Ecker bei Abbenrode

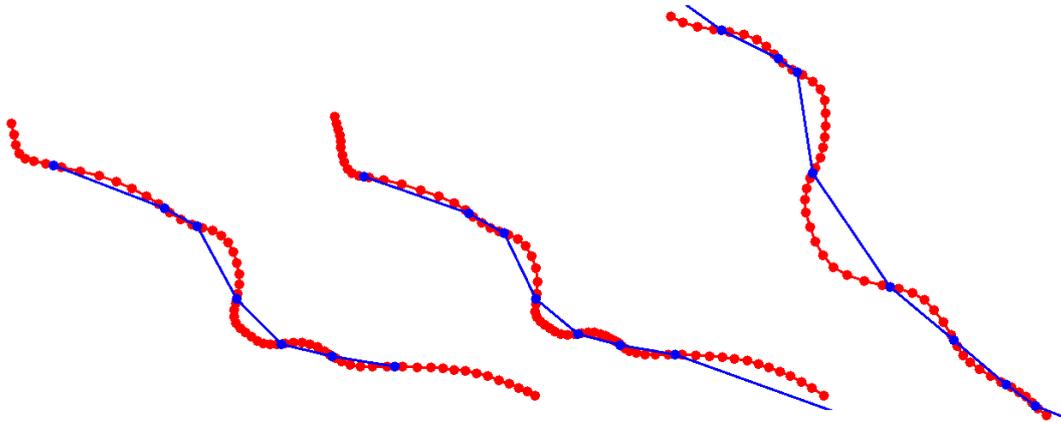


Abb. A.5.: Testabschnitt 5/5\*: Lindach in Kirchheim unter Teck und Goldach bei Trogen

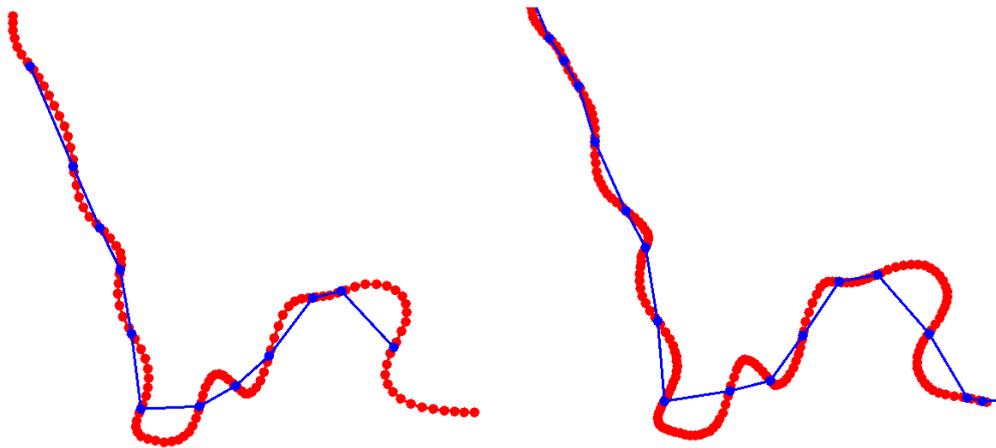


Abb. A.6.: Testabschnitt 6: Saale bei Saalfeld

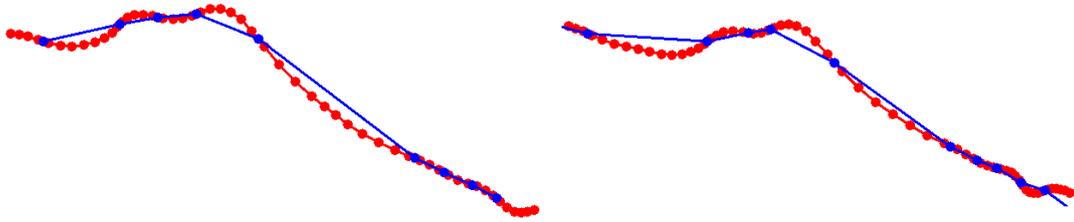


Abb. A.7.: Testabschnitt 7: Eschach bei Friesenhofen

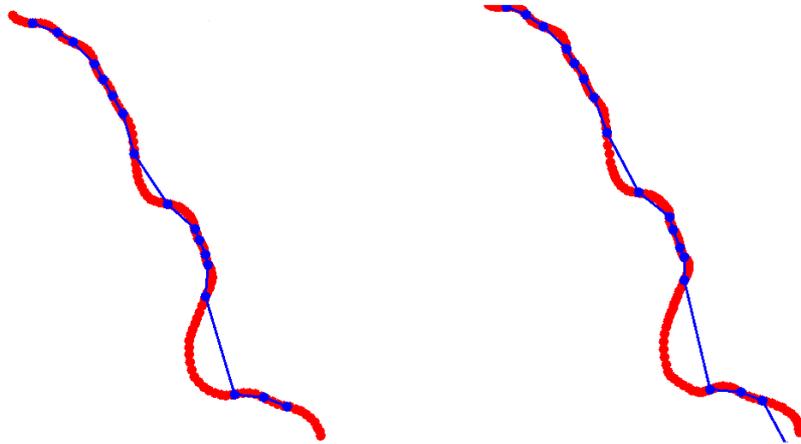


Abb. A.8.: Testabschnitt 8: Hunte bei Barnstorf

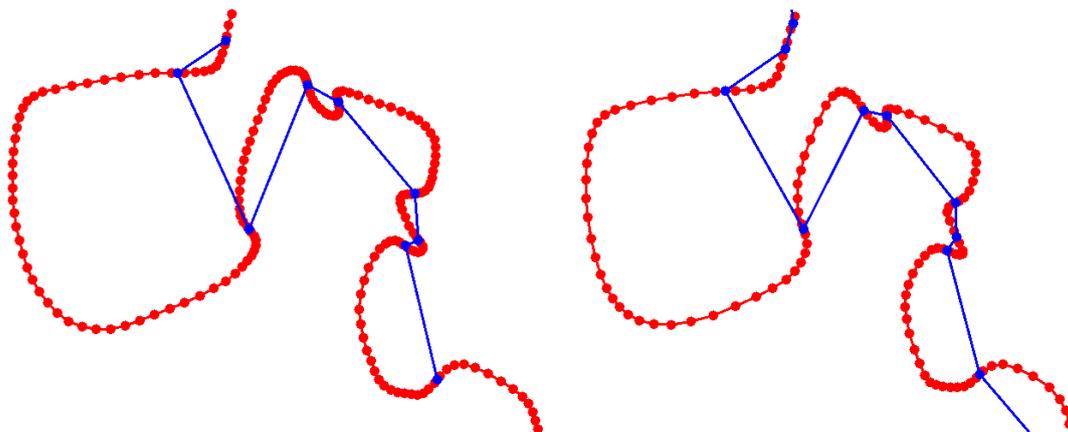


Abb. A.9.: Testabschnitt 9: Úhlava in Plzeň

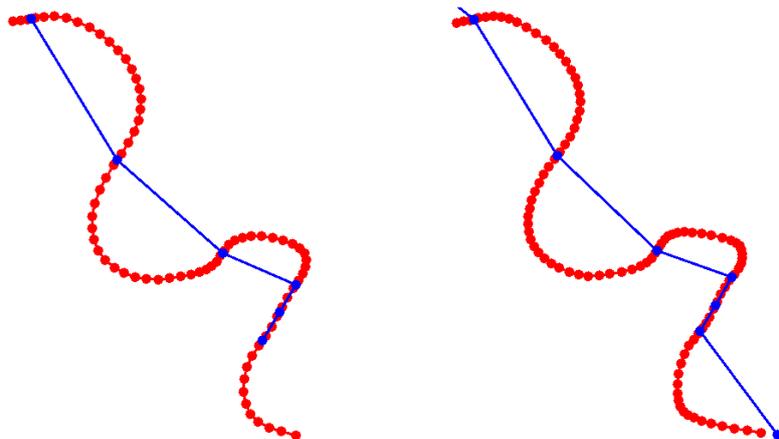


Abb. A.10.: Testabschnitt 10: Íssel bei Loikum

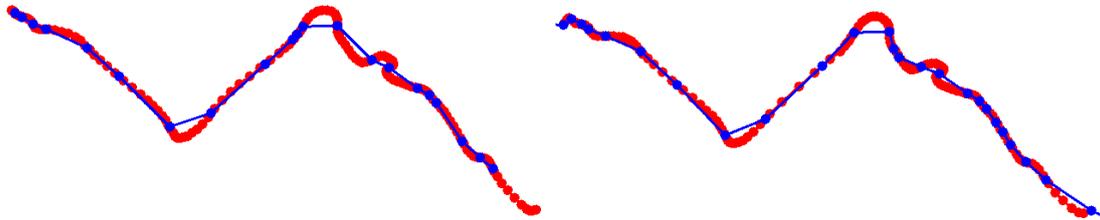


Abb. A.11.: Testabschnitt 11: Aller bei Osloss

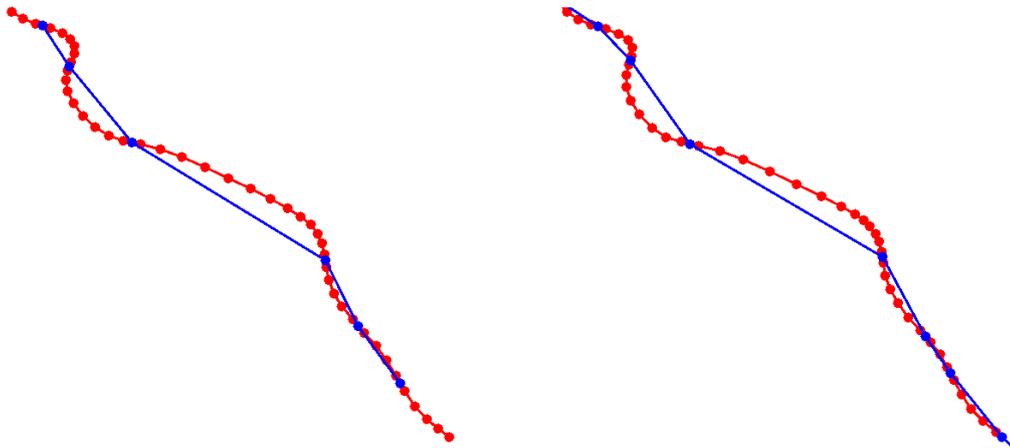


Abb. A.12.: Testabschnitt 12: Urft bei Urft

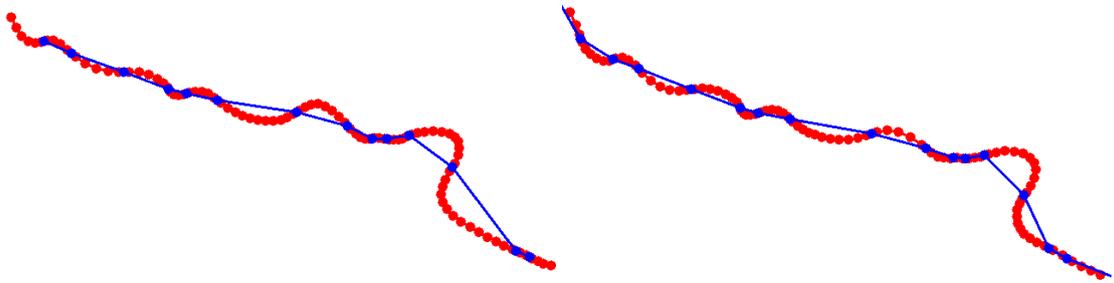


Abb. A.13.: Testabschnitt 13: Wyna bei Zetzwil

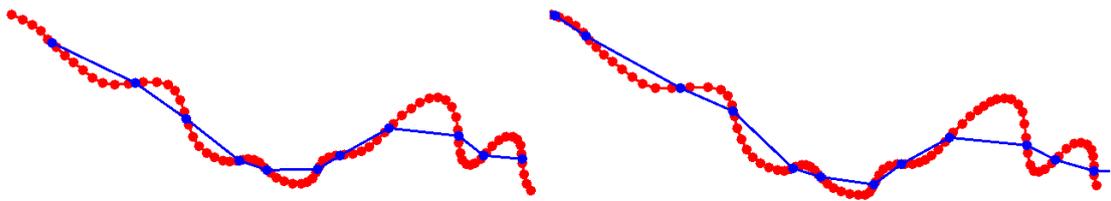


Abb. A.14.: Testabschnitt 14: Sorge bei Königsberg

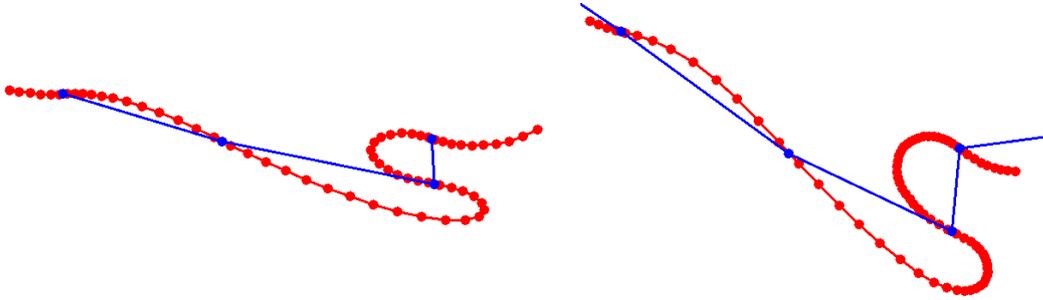


Abb. A.15.: Testabschnitt 15: Donau bei Schloggen

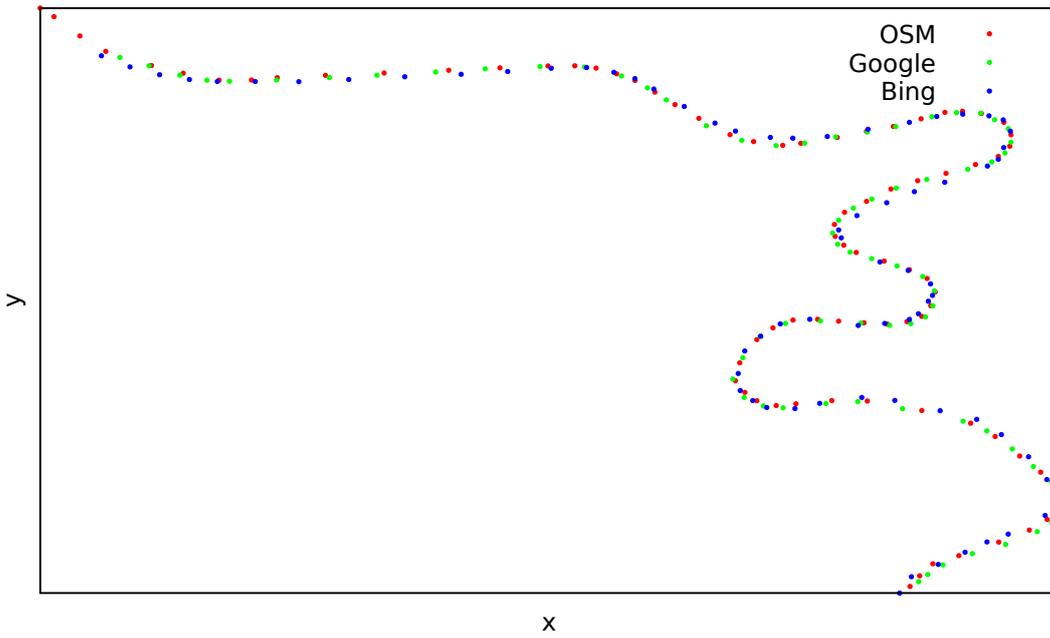


Abb. A.16.: Testabschnitt 3: manuell gesampelte Polygonzüge unterschiedlicher Quellen

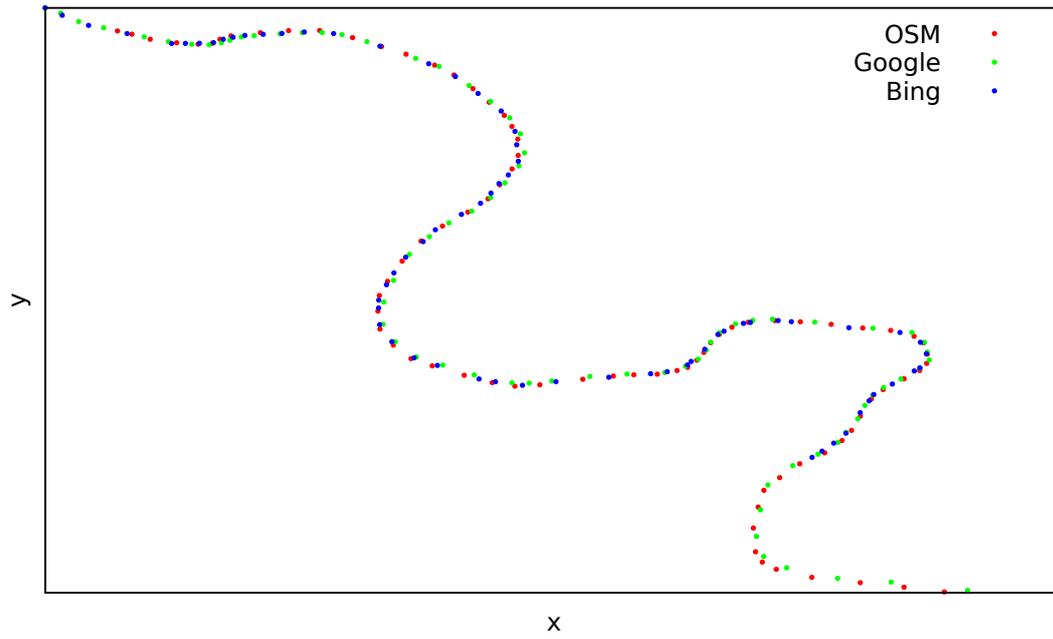


Abb. A.17.: Testabschnitt 10: manuell gesampelte Polygonzüge unterschiedlicher Quellen

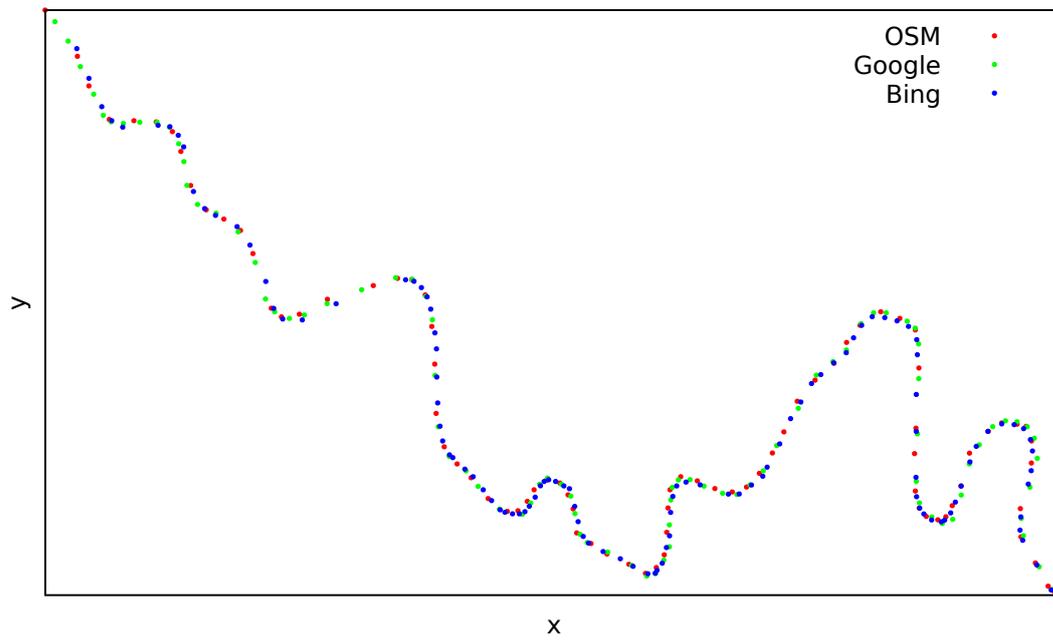


Abb. A.18.: Testabschnitt 14: manuell gesampelte Polygonzüge unterschiedlicher Quellen