



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

Ausarbeitung Anwendungen 2  
SoSe 2009  
Andrej Rull

Fahrspur- und Odometrie-basierte  
Selbstlokalisierung und Kartierung  
SLAM

## Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>2</b>
<b>1 Einführung</b>	<b>3</b>
<b>2 Das SLAM Problem</b>	<b>3</b>
<b>3 Related Works</b>	<b>4</b>
3.1 Visualization EKF-SLAM (vSLAM) . . . . .	5
3.2 Fast SLAM . . . . .	7
3.3 Bearing only SLAM . . . . .	9
3.4 ATLAS Framework . . . . .	11
<b>4 Abgrenzung zu eigenen Arbeitszielen und Methoden</b>	<b>13</b>
<b>5 Zusammenfassung</b>	<b>13</b>
<b>Literatur</b>	<b>14</b>

## Abbildungsverzeichnis

1	Unterschiedliche Koordinatenframes beim vSLAM <sup>1</sup> . . . . .	6
2	Systemzustand definiert durch mehrere Partikel . . . . .	7
3	Farhrzeugtrajektorie bei dem Partikel Filter <sup>2</sup> . . . . .	8
4	Omnidirektionale Kameraansicht <sup>3</sup> . . . . .	10
5	ATLAS Dijkstra-Projektion <sup>4</sup> . . . . .	11
6	Positionskorrektur durch benachbarte Frames <sup>5</sup> . . . . .	12

## 1 Einführung

Seit einigen Jahren werden im Automobilbereich autonomen mobilen Systemen erforscht. Dabei werden Systeme entwickelt, die den Menschen beim Führen eines Fahrzeugs unterstützen und insbesondere in kritischen Situationen in die Fahrzeugführung eingreifen sollen. Die Systeme werden in zwei Gruppen unterteilt, die Autonomen- und die Fahrassistenzsysteme. Während Fahrassistenzsysteme, wie bspw. ESP oder ABS, mittlerweile serienmäßig in den Fahrzeugen verbaut sind, wird an den vollautonomen Systemen noch geforscht.

Ein vollautonomes System ist in der Lage ein Fahrzeug ohne menschliches Eingreifen zu führen. Dabei werden Informationen der Umgebung über Sensoren aufgenommen, fusioniert und der Regelung des Fahrzeugs zu Verfügung gestellt. Damit eine globale Planung der Fahrt möglich ist, benötigt das Fahrzeug zusätzlich globale Informationen in Form einer Karte. Hat man am Anfang der Fahrt keine vorgegebene Karte, muss die Umgebung autonom kartiert werden, was sich in der Forschung als schwierig gestaltet. Diese Schwierigkeit wird im Kapitel 2. Genauer erläutert. Diese Ausarbeitung soll die beschriebene Problematik in Kapitel 3. aufnehmen und mehrere unterschiedliche Lösungsansätze vorstellen, die in der Forschung zu finden sind.

## 2 Das SLAM Problem

Simultaneous localization and mapping, kurz ausgedrückt SLAM, beschreibt das grundsätzliche Problem der autonomen Kartengenerierung. Steht am Anfang der Kartierung keine Umgebungsinformation dem Fahrzeug zu Verfügung muss es sich zunächst blind in der Umgebung bewegen. Die Position lässt sich dabei nur Anhand des kinematischen Modells des Fahrzeugs berechnen.

Durch Schlupf der Räder oder Unebenheiten auf der Strecke, schleichen sich Fehler in die Berechnung der Position ein, was zu fehlerhaften Kartierung der Umgebung führt. Korrigiert man die Fehler nicht, akkumuliert sich der Fehler in der Position mit der Zeit. Durch Kartendaten, versucht man deshalb Umgebungsmerkmale wieder zu erkennen und dadurch eine Korrektur durchzuführen. Da am Anfang jedoch keine Umgebungsinformationen vorhanden sind, ist die Korrektur nicht möglich, man spricht in der Literatur vom SLAM Problem.

Eine genaue Erläuterung des Problems und der Positionsbestimmung Anhand von Wegmessungen ist in [Rull (2008a)] nachzulesen.

### 3 Related Works

In der Forschung gibt es unterschiedliche Ansätze um das Simultaneous Location and Mapping Problem zu lösen. Die meisten Ansätze verwenden zur Lösung stochastische Methoden. Bei diesen Methoden wird die Fahrzeugposition  $p(x_p, y_p, \varphi)$ , sowie die Positionierung der Umgebungsmerkmale während der Fahrt geschätzt. Die Schätzung beruht auf dem Bewegungsmodell, in [ML. Benmessaoud (2008)] auch Prediction genannt, und wird später mit zusätzlichen Informationen über bekannte Muster der Umgebung optimiert (Updatephase). Die Umgebungsmuster müssen möglichst einen hohen Wiedererkennungswert haben, damit die Optimierung gewährleistet ist. Für die optimale Schätzung der globalen Fahrzeugkoordinaten und der Featurekoordinaten kommen zwei unterschiedliche Filter-Verfahren zum Einsatz.

Der erweiterte Kalman Filter (EKF), ist ein Single-Hypothesen-Ansatz und beschreibt das nicht lineare System durch einen Zustandsvektor, indem alle Koordinaten der Landmarken und die Positionskordinaten gespeichert sind. Diese werden zunehmend mit der Zeit aktualisiert und optimiert. Beim Partikel Filter (PF) wird die Position des Fahrzeugs durch ein Partikel beschrieben und die Landmarkenkoordinaten durch gering-dimensionierte EKFs. Der Partikel Filter ist anders als der EKF ein Multi-Hypothesen-Ansatz da alle Partikel in Konkurrenz zu einander stehen. Der Unterschied der beiden Filter wird in den weiteren Kapitel genauer erläutert, die Grundlagen sind in [Sebastian Thrun (2008)] nachzulesen.

[Diosi und L.Kleeman (2005)] setzen in ihrem Verfahren für die Wiedererkennungsmuster komplette Umgebungsscans anstelle von Landmarken ein. Die Definition lokaler Scans ist ebenfalls in [Ebert (2008)] und [Rull (2008b)] nachzulesen. Für scanbasierte Verfahren ist es wichtig Zyklen in der Umgebung während der Fahrt zu erkennen (Loop Closing), da der Fehler in der Positionierung nur durch Abgleich der Scans korrigiert werden kann. Den Abgleich der Scans bezeichnet man auch als Scan-Matching [Diosi und L.Kleeman (2005)].

Die Lösung des SLAM Problems wird durch zwei Modelle beschrieben. Während das Bewegungsmodell

$$p(x_t | x_{t-1}, u_t)$$

odometriebasierte Positionsbestimmung  $u_t$  implementiert, berechnet das messbasierte Modell

$$p(z_t | x_t, m)$$

zu einer aktuellen Position  $x_t$  im Weltmodell eine Messung  $z_t$ , die die aktuelle Umgebung am besten beschreibt. Dies kann sowohl ein Feature, als auch ein lokaler Scan sein. Die kartierte Umgebung ist mit  $m$  definiert.

Im Folgenden werden mehrere erforschte Verfahren vorgestellt, die die SLAM Problematik auf unterschiedlicher Art und Weise lösen.

### 3.1 Visualization EKF-SLAM (vSLAM)

In [ML. Benmessaoud (2008)] wird ein EKF-Verfahren vorgestellt, das ein Kamerabild als Sensor verwendet. Die Umgebungsmerkmale (Landmarken) werden nicht durch Entfernungsmesssystemen bestimmt, sondern aus dem Bild extrahiert. Kanten, Ecken und Regionen im Bild, definieren robuste Landmarken. Das Extrahieren erfolgt mittels dem Harris Corner detected Algorithmus.

Als Fahrzeug wird ein Roboter verwendet auf dem eine schwenkbare Kamera, sowohl in horizontaler als auch in vertikaler Richtung, angebracht ist. Kartiert wird eine Indoor Umgebung, da diese genug wiedererkennende Muster liefert. Der Vorteil einer schwenkbaren Kamera besteht darin, dass mehr Features in der Umgebung gefunden werden als bei einer statischen Kamera, was die Schätzung des Systemzustandes optimiert. Der Systemzustand zum Zeitpunkt  $k$  ist wie folgt definiert:

$$Y_k = \begin{pmatrix} R_k \\ \bar{L}^{g,k} \end{pmatrix}$$

Wo  $R_k = (x_k \ y_k \ z_k)^T$  die globale Position des Fahrzeugs darstellt, ausgegeben aus dem Bewegungsmodell und  $\bar{L}^{g,k} = (L_1 \ L_2 \ \dots \ L_N)$  ein Vektor aller globaler Landmarken  $L_i^{g,k} = (L_x^{g,k} \ L_y^{g,k} \ L_z^{g,k})^T$

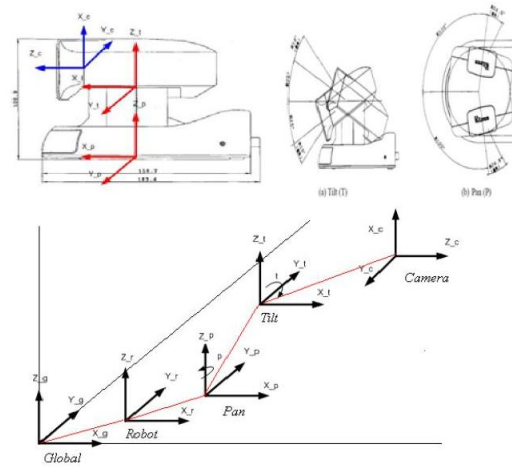
Für die Berechnung des Bewegungsmodells kommt die Odometrie zum Einsatz. Die Position des Fahrzeugs wird anhand von aktueller Wegmessung berechnet.

$$\begin{pmatrix} x(k+1) \\ y(k+1) \\ \phi(k+1) \end{pmatrix} = \begin{pmatrix} x(k) + v_k \cos \phi(k) T \\ x(k) + v_k \sin \phi(k) T \\ \phi(k) + w_k T \end{pmatrix}$$

Dabei gibt  $v_k$  die Translationsgeschwindigkeit in der Zeitperiode  $T$  an und  $w_k$  die Rotationsgeschwindigkeit.

Der Einsatz einer schwenkbaren Kamera führt zu mehreren 3-dimensionalen Koordinatenframes (Global, Roboter, Fahrzeug, Neigung, Schwenk, Kamera), dargestellt in [Abb. 1]. Damit Kartendaten mit den Bilderdaten abgeglichen werden können, werden die globalen Koordinaten der Landmarken  $L_i^{g,k} = (L_x^{g,k} \ L_y^{g,k} \ L_z^{g,k})^T$  in das kamerabasierte Koordinatensystem  $L_i^{c,k} = (L_x^{c,k} \ L_y^{c,k} \ L_z^{c,k})^T$  mittels einer Projektionsmatrix überführt. Die für die Berechnung verwendeten Projektionsmatrizen lassen sich in [ML. Benmessaoud (2008)] einsehen. Aus diesen Koordinaten werden entsprechend die Bildkoordinaten berechnet, wodurch das Beobachtungsmodell

$$I = h(R, Lg)$$

Abbildung 1: Unterschiedliche Koordinatenframes beim vSLAM <sup>1</sup>

zu einer Position und den globalen 3D-Koordinaten die entsprechenden Bildkoordinaten  $l = (u, v)$  liefert. Die Überführung von Bildkoordinaten in globale 3D Koordinaten, wird durch das Inverse Beobachtungsmodell bestimmt.

$$L_g = h^{-1}(R, l_1, R', l_2)$$

Wobei zwei Bilderframes  $l_1$  und  $l_2$  verwendet werden, aufgenommen von unterschiedlichen Positionen  $R, R'$ . Die Abarbeitung der Kartierung erfolgt in vier Schritten.

### 1. Initialisierung

Initialisierung des Systemzustandsvektors und der Kovarianzmatrix mit den Startkoordinaten  $R_0 = (0 \ 0 \ 0)^T$  des Roboters und den am Anfang erkannten Feature  $L_1 \dots L_N$ .

$$y_0 = (R_0 \ L_1 \ L_2 \ \dots \ L_N)^T$$

### 2. Prediction

Vorhersage des erwarteten Systemzustandes  $y_{k+1}$  unter Zunahme des alten Systemzustand

<sup>1</sup> [ML. Benmessaoud \(2008\)](#)

$y_k$  und der Odometrie  $u_k$  und Vorhersage der neuen Kovarianzmatrix  $\Sigma_{k+1}$ .

$$y_{k+1} = g(y_k, u_k)$$

### 3. Update

Korrigieren des Systemzustandes mittels des Beobachtungsmodells. Dabei ist  $I_k$  die aktuell aufgenommene Landmarke in der Umgebung und  $\hat{I}_k$ , die vom Beobachtungsmodell ausgegebene Landmarke. Durch die Kalman Verstärkung  $Ka$  wird die Schätzungs des neuen Zustandes optimiert.

$$y_{k+1} = y_{k+1} + Ka(I_k - \hat{I}_k)$$

### 4. Map Management

Gibt es kein Zusammenhang zwischen einer aktuell aufgenommenen Landmarke und einer Landmarke im Systemzustandsvektor, so erweitert das Map Management den Zustandsvektor und die Kovarianzmatrix um die neue Landmarke. Die globalen Koordinaten der Landmarke werden wie oben erwähnt durch das Inverse-Beobachtungsmodell bestimmt.

### 3.2 Fast SLAM

Anders als beim vSLAM Verfahren, das den kompletten Systemzustand mittels einer Hypothese schätzt (EKF), wird in [M. Montemerlo (2002)][M. Montemerlo (2003)][J.Z. Sasiadek (2008)] ein Verfahren verwendet, das für die Schätzung des Systemzustandes mehrere konkurrierende Hypothesen nutzt. Dabei wird der Nachteil des Single-Hypothesen Ansatzes, bei dem die Kovarianzmatrix quadratisch  $O(N^2)$  in Abhängigkeit der Landmarkenanzahl  $N$  anwächst und dadurch der Rechenaufwand des Filters quadratischen zunimmt, durch einen Partikel Filter kompensiert.

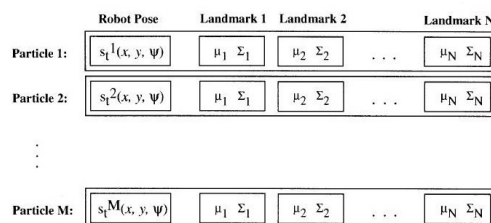


Abbildung 2: Systemzustand definiert durch mehrere Partikel

Der Partikel Filter (PF) beschreibt die gefahrene Trajektorie des Fahrzeugs in Verbindung zur Karte [Abb. 3]. Das System wird nicht mehr durch einen Zustandsvektor beschrieben, sondern durch Partikel die eigene Hypothesen über den Systemzustand aufstellen. Dafür enthalten die Partikel die Positionskordinaten in der Umgebung und eine eigene Karte, bestehend aus  $N$  klein-dimensionierten EKFs (für die Schätzung der Landmarkenkoordinaten), wie in [Abb. 2] dargestellt. Durch den Aufbau reduziert sich der Speicheraufwand beim PF auf  $O(NM)$ , wobei  $N$  die Anzahl der Landmarken darstellt und  $M$  die der Partikel. Der Zeitaufwand beträgt bei der Aktualisierung des PF laut M. Montemerlo (2003)  $O(M \log N)$ .

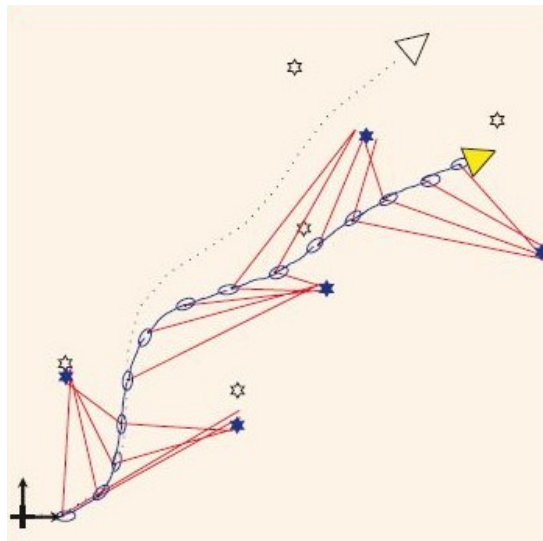


Abbildung 3: Fahrzeugtrajektorie bei dem Partikel Filter <sup>2</sup>

Die Systemzustand (Landmarken  $\theta$  und Pfad  $s^t$ ) zu einem Zeitpunkt  $t$ , wird mittels den Odometriedaten  $u^t$ , den Umgebungsmessungen  $z^t$  und den observierten Landmarken  $n^t$  zum Zeitpunkt  $t$  berechnet.

$$p(\theta, s^t | z^t, u^t, n^t) = p(s^t | z^t, u^t, n^t) \prod p(\theta_n | s^t, z^t, u^t, n^t)$$

Dabei gibt der Erste Term das Bewegungsmodell und der zweite Beobachtungsmodell an. Das FAST SLAM Verfahren wird insbesondere für große Outdoor-Umgebungen entwickelt, in denen die reinen EKF-basierten SLAM Verfahren wegen der hohen Landmarkenanzahl scheitern. Für die Umgebungsmessung kommen Entfernungsmesssysteme, wie der Laserscanner, zum Einsatz.

<sup>2</sup> Durrant-Whyte und Bailey (2006)



### 3.3 Bearing only SLAM

Wird bei der Kartierung eine Kamera als Sensor verwendet, hat man die Problematik der fehlenden Distanzinformationen bzw. der fehlenden Tiefeninformationen im Bild. Ohne diese Informationen ist es schwierig eine Positionierung im Raum abzuschätzen und so Landmarken im Raum zu definieren. Das Bearing only SLAM Verfahren gilt in der Literatur als mögliche Lösung solch eines Problems. Dabei extrahieren Verfahren spezielle Keypoints aus den Bilddaten, die gegen Translation und Skalierung konsistent sind. Um die Tiefeninformation zu erhalten werden in [T. Lemaire (2005)] zu diesen Punkten dreidimensionale Richtungsvektoren abgebildet. Eine Gleichverteilung von Gaußfunktionen entlang des Vektors repräsentiert die Tiefeninformation. Durch mehrere fortlaufende Bilder überlagern sich die Vektoren mit den gleichverteilten Gaußfunktionen, wodurch sich die Feature-Positionen approximieren lassen.

#### Bearing only SLAM mit einem PF

[Hauke Strasdat (2007)] setzt eine auf dem Roboter festmontierte monokulare Kamera als Bewegungssensor ein. Der Speeded-up visual feature (SURF) Algorithmus, der in [Herbert Bay (2008)] beschrieben wird, definiert robuste Punkte im Bild. Für die Auffindung der Punkte dient der Fast-Hessian Detector, welcher auf der Approximation der Hessematrix eines Bildes basiert. Im Messmodell werden die SURF Feature in sphärischen Koordinaten angegeben.

$$z = (\phi \theta \rho)^T$$

Die benötigten Winkel (Polarwinkel  $\phi$ , Azimutwinkel  $\theta$ ) werden durch Pixelkoordinaten im Bild bestimmt. Die Entfernung  $\rho$  zum Feature wird zunächst ignoriert, da sie nicht berechnet werden kann. Später wird diese wie oben erwähnt mit Hilfe mehrerer Bilder und einer Vektorüberlagerung approximiert. Für die Schätzung des Systemzustandes (Fahrzeugposition, sowie globale Landmarkenpositionen) wird der Partikel Filter verwendet. Jede Landmarke  $\mu = (x \ y \ z)^T$  wird durch einen eigenständigen EKF  $(\mu, \Sigma)$  modelliert. Für den Abgleich von gespeicherten Landmarken, werden die globalen Kartenkoordinaten  $\mu$  in sphärische Kamerakoordinaten  $z = (\phi \ \theta \ \rho)^T$  überführt. Das Verfahren ermöglicht dadurch ein Abgleich der Landmarken ohne Distanzinformation. Die Wahrscheinlichkeit des Zusammenhangs einer gespeicherten Landmarke und einer neuen Messung, wird nur mittels der zwei Winkel berechnet. Wird die Beobachtungswahrscheinlichkeit der Übereinstimmung als zu gering bewertet, kommt es zur Neudefinition der Landmarken, indem die Tiefeninformation  $\rho$  über mehrere Bilder approximiert und die sphärischen Kamerakoordinaten in kartesische Kartenkoordinaten projiziert werden. Auch wird die Beobachtungswahrscheinlichkeit verwendet um die Partikel im System zu gewichten, wodurch die optimale Roboterposition auf dem Pfad geschätzt wird.

### Bearing only SLAM mit einem EKF

Das Verfahren in [[Siegfried Hochdorfern \(2007\)](#)] verwendet für die Systemschätzung nur einen erweiterten Kalman Filter, der mit unechten Landmarken umgehen kann. Die Landmarken werden durch den Scale Invariant Feature Transform (SIFT) Algorithmus definiert. Eine omnidirektionale Kamera dient als Sensor. Die Kamera hat den Vorteil, dass man durch einen hyperbolischen Spiegel einen 360 Grad Blickwinkel erhält, zu sehen in [Abb. 4]. Ein neu extrahiertes SIFT Feature wird zunächst mit den Features in einer Datenbank abgeglichen und dadurch klassifiziert (neue Landmarke, nicht initialisierte Landmarke, initialisierte Landmarke). Die dabei erhaltene ID, der Peilungswinkel zu der Landmarke und die Odometriedaten des Roboters stellen dann den Systeminput für den Kalman-Filter dar.

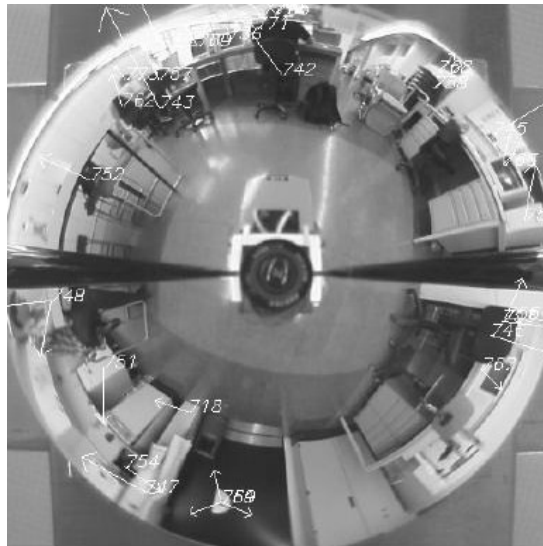


Abbildung 4: Omnidirektionale Kameraansicht <sup>3</sup>

<sup>3</sup> [Siegfried Hochdorfern \(2007\)](#)

### 3.4 ATLAS Framework

Um insbesondere mit der Kartierung großer Umgebungen umgehen zu können, wird in [Michael Bosse (2003)][Michael Bosse (2004)] und [Michael Bosse (2008)] ein Framework mit einem schmal-skaliertem Algorithmus eingesetzt, was sowohl feature- als auch scanbasiert arbeitet. Vergleichbar mit einem Partikel Filter handelt es sich beim ATLAS Framework ebenfalls um ein Multi-Hypothesen Ansatz. Das kartierte Weltmodell wird nicht in einem globalen Koordinatensystem, sondern in einem Graph von lokalen Frames, die ein eigenes lokales Koordinatensystem implementieren, repräsentiert.

Dabei enthält ein lokaler Frame (Knoten im Graph) neben den Messdaten eines kartierten Umgebungsbereichs und der Framepositionierung, eine Hypothese über den Zustand. Diese bewertet den Zusammenhang zwischen Framedaten und den aktuellen aufgenommenen Messungen. Der Frame, der eine aktuelle Umgebungsmessung am besten beschreibt, wird aktiviert und aktualisiert. Bei der featurebasierten Lösung implementieren die Knoten dabei einen EKF, bei den scanbasierten Verfahren die Scan-Match Algorithmen wie bspw. den ICP Algorithmus [Lu und Milios (1997)].

Frames haben eine festdefinierte Größe, ein Stammprozess entscheidet ob in ein benachbartes Frame übergegangen wird oder ob ein neues erzeugt wird. Kanten, die die Frames miteinander verbinden, repräsentieren Transformationen zwischen den Frames. Wird ein neuer Knoten in den Graph aufgenommen, wird nach anderen Knoten im Graph gesucht, die die Eckpunkte verknüpfen. Da mehrere Pfade zu dem jeweiligen Knoten existieren können, wird der Dijkstra Shortest Path Algorithmus verwendet um die Doppeldeutigkeit zu beheben.

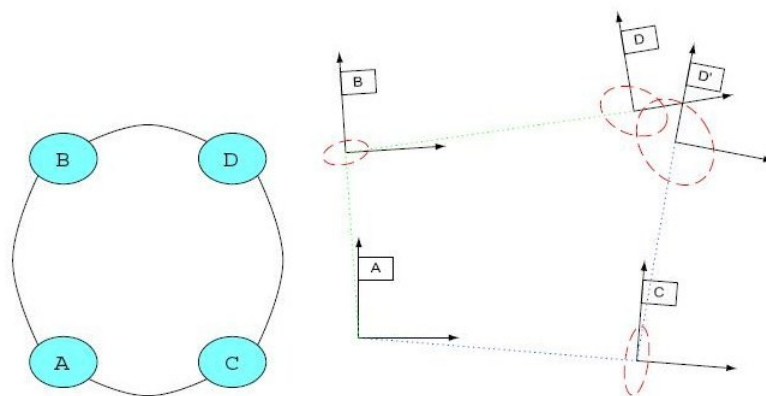


Abbildung 5: ATLAS Dijkstra-Projektion <sup>4</sup>

<sup>4</sup> Michael Bosse (2004)

Abb. zeigt ein Beispiel für eine Doppeldeutigkeit der Transformationen  $T_a^d$  von Knoten A zum Knoten D. Der kleinste Pfad ist durch den Knoten B gegeben, was durch die kleinere Fehlerellipse angezeigt wird. Das heißt Metrik  $p$  der Transition  $T_a^c$  ist besser als der Transition  $T_a^b$ . Für die Dijkstra-Projektion wird eine statistische Metrik  $p$  benutzt, die auf Wahrscheinlichkeit in den Transformationen der Kanten beruht. Die Metrik  $p$  ist die Determinante der Kovarianzmatrix der zusammengesetzten Transformation.

$$T_a^d = T_a^b \oplus T_b^d$$

$$\sum_{ad} = J_1(T_a^b, T_b^d) \sum_{ab} J_1(T_a^b, T_b^d)^T + J_2(T_a^b, T_b^d) \sum_{bd} J_2(T_a^b, T_b^d)^T$$

$$p = \det(\sum_a d)$$

Die Zusammensetzung der Transformation lässt sich mittels dem  $\oplus$  Operator berechnen. Für die Berechnung der Kovarianzmatrix  $\sum_{ad}$  werden die Jacobi Matrizen  $J_1$  und  $J_2$  verwendet, die die Ableitungsmatrizen der Transformationsfunktionen angeben.

Die Positionsschätzung und Aktualisierung im Frame und somit der Fahrzeugposition erfolgt bei der landmarkenbasierten Lösung durch den EKF im Frame. Bei scanbasierter Implementierung erfolgt die Aktualisierung durch benachbarte Frames. [Abb. 6] veranschaulicht die Aktualisierung der Position. Die aktuelle Position ist durch  $x_j$  gekennzeichnet. Der benachbarte Frame  $i$  wird nun aufgefordert seine Position  $x_i^{old}$  entsprechend zu aktualisieren  $x_i^*$ .

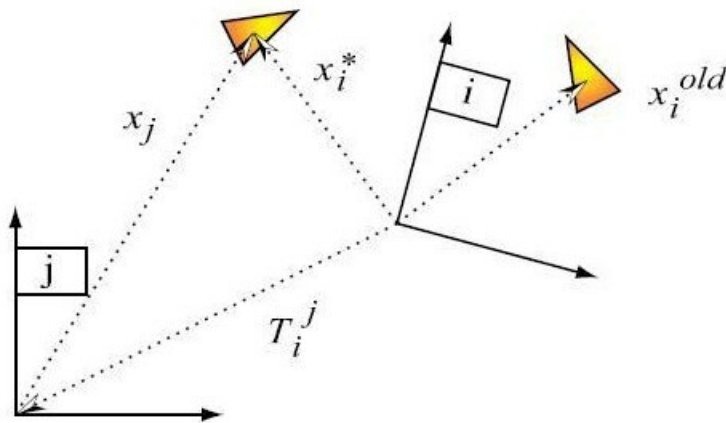


Abbildung 6: Positionskorrektur durch benachbarte Frames <sup>5</sup>

<sup>5</sup> Michael Bosse (2004)

## 4 Abgrenzung zu eigenen Arbeitszielen und Methoden

Im Bezug auf die eigene Problemstellung, wo eine aufgeklebte Strecke mittels eines Modellfahrzeugs kartographiert werden soll, können die vorgestellten Verfahren nur bedingt angewandt werden, da die Einsatzumgebung nicht genug Informationen bietet um eindeutige Feature zu extrahieren. Die eigene Lösung basiert deshalb auf ein scanbasiertes Multi-Hypothesen-Modell, was als Sensorinput das Polaris System verwendet, was in [\[Jenning \(2008\)\]](#) entwickelt wurde.

Der lokale Frameansatz des ATLAS Frameworks soll verwendet werden um Streckenbereiche in Frames zu unterteilen. Diese Bereiche werden Section-Frames genannt und werden wie beim ATLAS in einem Graphen abgelegt. Der Abgleich gespeicherter Frames mit den neuen Messungen erfolgt durch Scan-Matching. Wird ein Zyklus festgestellt, werden die Positionen in den Frames rekursive korrigiert. Da sich einige Bereiche der Strecke ähneln können, wodurch eine aktuelle Messung einem vorhandenen Frame nicht eindeutig zugewiesen werden kann, entscheidet ein Stammprozess über die Aktivierung der Section-Frames.

## 5 Zusammenfassung

In dieser Ausarbeitung wurden unterschiedliche Verfahren zur Lösen der SLAM Problematik vorgestellt. Stochastische Filter, wie der erweiterte Kalman-Filter und der Partikel-Filter bilden beim featurebasierten Verfahren die Grundlage für die Schätzung des Systemzustandes, der die Fahrzeugposition und die Landmarkenpositionen angibt.

Für die Featureextraktion können sowohl Entfernungsmesssysteme, als auch visuelle Kamerabilder verwendet werden. Bei kamerabasierte Verfahren werden für die Suche eindeutiger Featurepunkte die Algorithmen SIFT und SURF eingesetzt.

Alle Arbeiten befassen sich ausschließlich mit der Innenraum- bzw. der Outdoor-Umgebungskartierung. Dabei bietet die Umgebungen genug Information um Landmarken zu extrahieren um dadurch die Schätzung der Fahrzeugposition nach einer Zeit zu optimieren. Stellt die Umgebung jedoch keine eindeutigen Landmarken zur Verfügung, scheitern die erwähnten Verfahren, wodurch ein eigener Ansatz für die Lösung des SLAM Problems bei einer Streckenkartierung entwickelt werden muss.

## Literatur

- [Arnaud Doucet 2000] ARNAUD DOUCET, Kevin Murphy Stuart R.: Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks. (2000)
- [Diosi und L.Kleeman 2005] DIOSI, A. ; L.KLEEMAN: Laser scan matching in polar coordinates with application to SLAM. In: *IEEE International Conference on Intelligent Robots and Systems* (2005), Aug, S. 3317–3322
- [Durrant-Whyte und Bailey 2006] DURRANT-WHYTE, Hugh ; BAILEY, Tim: Simultaneous Localization and Mapping: Part I. In: *IEEE Robotics and Automation Magazine* (2006), S. 99–108
- [Ebert 2008] EBERT, Michael: *Aktives Mapping und Positionsbestimmung eines autonomen Modellfahrzeugs*, Hochschule für Angewandte Wissenschaften Hamburg, Bachelorarbeit, 2008
- [Hauke Strasdat 2007] HAUKE STRASDAT, Maren Bennewitz Wolfram B.: Visual Bearing-Only Simultaneous Localization and Mapping with Improved Feature Matching. In: *Autonome Mobile Systeme 2007, 20. Fachgespräch, Kaiserslautern* (2007), S. 15–21
- [Herbert Bay 2008] HERBERT BAY, Tinne Tuytelaars Luc Van G.: SURF: Speeded Up Robust Features. In: *Computer Vision and Image Understanding (CVIU)* 110 (2008), S. 346–359
- [Jenning 2008] JENNING, Eike: *Systemidentifikation eines autonomen Fahrzeugs mit einer robusten, kamerabasierten Fahrspurerkennung in Echtzeit*, Hochschule für Angewandte Wissenschaften Hamburg, Masterarbeit, 2008
- [J.Z. Sasiadek 2008] J.Z. SASIADEK, D. N.: Navigation of an autonomous mobile robot using EKF-SLAM and FastSLAM. In: *Mediterranean Conference on Control and Automation* 16 (2008), Juni, S. 517–522
- [Lu und Milios 1997] LU, Feng ; MILIOS, Evangelos: Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans. In: *Journal of Intelligent and Robotic Systems* 18 (1997), S. 249–275
- [M. Montemerlo 2002] M. MONTEMERLO, D. Koller B. W.: FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem. In: *Proceedings of the AAAI National Conference on Artificial Intelligence* (2002), S. 593–598
- [M. Montemerlo 2003] M. MONTEMERLO, D. Koller B. W.: FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In: *Proceeding of International Joint Conferences on Artificial Intelligence* (2003), S. 1151–1156

- [Michael Bosse 2003] MICHAEL BOSSE, John Leonard Seth T.: An atlas framework for scalable mapping. (2003), S. 1899–1906
- [Michael Bosse 2004] MICHAEL BOSSE, John Leonard Seth T.: Simultaneous localization and map building in large-scale cyclic environments using the Atlas Framework. 23 (2004), S. 1113–1139
- [Michael Bosse 2008] MICHAEL BOSSE, John Leonard Seth T.: Map Matching and Data Association for Large-Scale Two-dimensional Laser Scan-based SLAM. 27 (2008), S. 667–691
- [ML. Benmessaoud 2008] ML. BENMESSAOUD, K. Nemra AK. S.: Single-Camera EKF-vSLAM. In: *International Journal of Intelligent Systems and Technologies* 3 (2008)
- [N. M. Kwok 2008] N. M. KWOK, G. D.: Bearing-only SLAM in Indoor Environments Using a Modified Particle Filter. (2008)
- [Rull 2008a] RULL, Andrej: *Fahrspur- und Odometrie-basierte Selbstlokalisierung und Kartierung (SLAM) - Problemstellung*, Hochschule für Angewandte Wissenschaften Hamburg, Ausarbeitung, 2008. – URL [http://users.informatik.haw-hamburg.de/~rull\\_a/download/Ausarbeitung\\_AW1\\_Andrej\\_Rull.pdf](http://users.informatik.haw-hamburg.de/~rull_a/download/Ausarbeitung_AW1_Andrej_Rull.pdf)
- [Rull 2008b] RULL, Andrej: *Sensorbasierte Umgebungskartierung mit lokaler Positionskorrektur für autonome Fahrzeuge*, Hochschule für Angewandte Wissenschaften Hamburg, Bachelorarbeit, 2008
- [Sebastian Thrun 2008] SEBASTIAN THRUN, John J. L.: Simultaneous Localization and Mapping. In: *Handbook for Robotics* (2008), S. 871–889
- [Siegfried Hochdorfern 2007] SIEGFRIED HOCHDORFERN, Christian S.: Bearing-Only SLAM with an Omnicam Robust Selection of SIFT Features for Service Robots. In: *Autonome Mobile Systeme 2007, 20. Fachgespräch, Kaiserslautern* (2007), S. 8–14
- [T. Lemaire 2005] T. LEMAIRE, J. S.: A practical 3D bearing-only SLAM algorithm. In: *International Conference on Intelligent Robots and Systems* (2005), S. 2449–2454