

ERWEITERUNG EINES KI-GESTÜTZTEN ASSISTENZSYSTEMS ZUR OPTIMIERUNG VON LASERSCHNEIDPARAMETERN FÜR EDELSTAHLBLECHE

Projektarbeit T2000

des Studienganges Elektrotechnik
Fachrichtung Automation
an der Dualen Hochschule Baden-Württemberg
Standort Stuttgart

Tudor Lupsa

08.09.2025

Bearbeitungszeitraum	02.06.25 - 08.09.25
Matrikelnummer, Kurs	1491114, TEL23GR3
Dualer Partner	TRUMPF SE+Co.KG, Ditzingen
Betreuer des Dualen Partners	Manuel Geiger, M.Sc

Sperrvermerk

Die vorliegende Projektarbeit beinhaltet interne und vertrauliche Informationen der Firma TRUMPF SE + Co. KG. Die Weitergabe des Inhalts, der Arbeit im Gesamten oder in Teilen, sowie Anfertigen von Kopien oder Abschriften, auch in digitaler Form, sind grundsätzlich untersagt. Ausnahmen bedürfen der schriftlichen Genehmigung durch Herrn Manuel Geiger, Betreuer dieser Projektarbeit bei TRUMPF SE + Co. KG Ditzingen.

Dieser Sperrvermerk gilt zum 13. Januar 2026.

Eidesstattliche Erklärung

Hiermit versichere ich, die vorliegende Projektarbeit selbstständig und nur unter Verwendung der von mir angegebenen Quellen und Hilfsmittel verfasst zu haben. Sowohl inhaltlich als auch wörtlich entnommene Inhalte wurden als solche kenntlich gemacht. Die Arbeit hat in dieser oder vergleichbarer Form noch keinem anderem Prüfungsgremium vorgelegen.

Datum: _____ Unterschrift: _____

Kurzreferat

Abstract

Inhaltsverzeichnis

Abkürzungsverzeichnis	V
Abbildungsverzeichnis	VI
Tabellenverzeichnis	VII
Quellcodeverzeichnis	VIII
1 Einführung	1
1.1 Zielsetzung	1
1.2 Vorgehensweise	1
1.3 Anwendungsszenario Lageschätzung	2
2 Grundlagen und Stand der Technik	3
2.1 Rapid Control Prototyping	3
2.2 MBD mit Matlab/Simulink	3
2.3 Quaternionen und Euler-Winkel	4
2.4 Lageschätzung mittels Sensorfusion	7
2.4.1 Erweitertes Kalmanfilter	8
2.4.2 Komplementärfilter	10
Literaturverzeichnis	12

Abkürzungsverzeichnis

RCP	Rapid Control Prototyping
IMU	Inertial Measurement Unit, deutsch: Inertialmesseinheit
KF	Kalmanfilter

Abbildungsverzeichnis

1	Schematische Darstellung der Euler-Winkel	4
2	Blockdiagramm zur Veranschaulichung der Funktionsweise eines Komplementärfilter	10

Tabellenverzeichnis

Quellcodeverzeichnis

1 Einführung

Dieses Kapitel gibt eine Einführung in die Thematik der Projektarbeit. Es werden die Zielsetzung und die geplante Vorgehensweise beschrieben.

1.1 Zielsetzung

Das Ziel dieser Projektarbeit ist es eine Versuchsplattform für die Automatisierungstechnik zu entwickeln bei der typische Regelungstechnische Methoden experimentelle untersucht werden und anschließend auf die Zielplattform übertragen werden können. Hierfür soll ein cro:RCPRapid Control Prototyping (RCP)-System verwendet werden, um die Regelungsalgorithmen in Echtzeit auf der Zielplattform auszuführen, indem bereits vorhandene Hardware und Software Bausteine genutzt werden. Das RCP ist ein Verfahren mit dem zu regelnde Systeme schnell und flexibel entwickelt und getestet werden können. Hierbei ist es nicht notwendig manuelle Implementierung in Programmiersprachen für die Zielhardware zu erstellen, sondern es kann direkt von einer grafischen Simulationsumgebung wie z.B. *Matlab/Simulink* auf die Zielhardware übertragen werden mithilfe von einer automatischen Codegenerierung. Dies ermöglicht eine schnelle Iteration und Anpassung der Regelungsalgorithmen, was besonders in der Entwicklungsphase von Vorteil ist, da somit schnell und kosteneffizient Prototypen erstellt und getestet werden können. Die Versuchsplattform soll einen durchgängigen Prozess von der Modellerstellung über die Simulation bis zur Echtzeitausführung auf der Zielhardware abbilden und dabei die experimentelle Parametrierung, Optimierung sowie die Beobachtung relevanter Signale und Messgrößen ermöglichen, indem eine Kopplung zwischen Entwicklungsrechner und Zielplattform zur Signal und Parameterkommunikation genutzt wird (Hoyos-Gutiérrez et al. 2023).

1.2 Vorgehensweise

Die Projektarbeit setzt auf eine Einarbeitung in das Rapid Control Prototyping, um die theoretischen Grundlagen, die verwendeten Begriffe und typische Prozessabläufe einzuordnen und daraus geeignete Vorgehensprinzipien abzuleiten. Aufbauend auf diesem Kenntnisstand werden Anforderungen an die zu entwickelnde Versuchsplattform definiert. Dabei werden funktionale Anforderungen beschrieben, die sich aus den geplanten Experimenten ergeben, sowie nicht funktionale Anforderungen festgelegt, die unter anderem die Umsetzbarkeit, Erweiterbarkeit und Randbedingungen der Nutzung betreffen.

Nachdem die Randbedingungen gesetzt wurden, erfolgt die Auswahl geeigneter Hardware und Software Werkzeuge für die Umsetzung der Versuchsplattform. Ziel ist es, eine Kombination

aus Hardware und Software zu identifizieren, die eine effiziente Entwicklung, Simulation und Echtzeitausführung der Regelungsalgorithmen ermöglicht.

Parallel zum physischen Aufbau wird die Versuchsplattform in Matlab und Simulink modelliert, um ein ausführbares Systemmodell für die Simulation bereitzustellen. In diesem Modell werden Sensorik, Aktorik und die wesentlichen dynamischen Eigenschaften des Prozesses abgebildet, sodass Algorithmen vor der Implementierung auf der realen Plattform unter definierten Bedingungen untersucht werden können. Abschließend wird der Ansatz des Rapid Control Prototyping exemplarisch angewendet, indem ausgewählte Funktionen in Simulink entworfen, simuliert und anschließend mittels automatischer Codegenerierung auf die Zielhardware übertragen werden. Die Implementierung wird getestet und validiert, um die Funktionalität und Leistungsfähigkeit der entwickelten Versuchsplattform zu gewährleisten.

Für die Umsetzung der RCP-Methodik wird in der Projektarbeit ein Anwendungszenario der "Lageschätzung" genutzt. Hierbei soll eine cro:IMU Inertial Measurement Unit, deutsch: Inertialmesseinheit (IMU) simuliert werden und gegebenenfalls aus der ausgewählten Hardware eine reale IMU ausgelesen werden. Anschließend soll die Lage des Systems in Form von Quaternionen/Eulerwinkeln geschätzt werden (HIER VERWEIS KAPITEL 2.4). Die Lageschätzung soll auf Grundlage einer Sensorfusion funktionieren, indem Beschleunigungs- und Gyroskopdaten kombiniert werden. Hierfür sollen verschiedene Algorithmen implementiert und getestet werden, um die Genauigkeit und Robustheit der Lageschätzung zu bewerten. Ziel ist es, eine zuverlässige Methode zur Bestimmung der Systemlage zu entwickeln, die in Echtzeit auf der RCP-Plattform ausgeführt werden kann.

1.3 Anwendungsszenario Lageschätzung

NilsNils

2 Grundlagen und Stand der Technik

2.1 Rapid Control Prototyping

"Tudor"

2.2 MBD mit Matlab/Simulink

"Tudor"

2.3 Quaternionen und Euler-Winkel

Quaternionen und Euler-Winkel sind mathematische Darstellungen, um Rotationen im dreidimensionalen Raum zu beschreiben. Solche Rotationsbeschreibungen werden unter anderem in der Lageschätzung von Inertialmesseinheiten (IMU) benötigt, um die Orientierung aus Sensormessgrößen zu modellieren und zu bewerten.

Grundsätzlich kommen hierfür sowohl Quaternionen als auch Euler-Winkel in Frage. Quaternionen eignen sich besonders für die interne Berechnung, da sie eine kompakte und singularitätsfreie Darstellung von Rotationen ermöglichen. Euler-Winkel werden hingegen häufig zur Ausgabe und Visualisierung verwendet, da sie die Orientierung anschaulich über drei aufeinanderfolgende Winkel beschreiben: *Yaw* (Gierwinkel), *Pitch* (Nickwinkel) und *Roll* (Rollwinkel). Abbildung 1 zeigt diese Rotationen um die körperfesten Achsen. *Roll* entspricht einer Rotation um die x -Achse, *Pitch* einer Rotation um die y -Achse und *Yaw* einer Rotation um die z -Achse.

Euler-Winkel besitzen jedoch Singularitäten und sind daher als interne Zustandsdarstellung nur eingeschränkt geeignet; in der Praxis werden sie meist für die Visualisierung der geschätzten Orientierung verwendet.

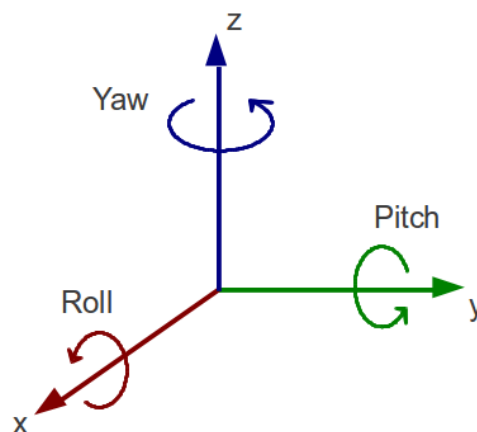


Abb. 1: Schematische Darstellung der Euler-Winkel (Iman Bagheri 2026)

Für die konkrete mathematische Beschreibung ist zu beachten, dass Euler-Winkel stets an eine festgelegte Rotationskonvention gebunden sind. Da räumliche Rotationen nicht vertauschbar sind, führt eine andere Reihenfolge der Einzelrotationen zu einer anderen resultierenden Orientierung. Beispielsweise liefert die Abfolge *Yaw–Pitch–Roll* im Allgemeinen ein anderes Ergebnis als *Pitch–Yaw–Roll*, selbst wenn die gleichen Winkelwerte verwendet werden.

Neben dieser Konventionsabhängigkeit besitzen Euler-Winkel einen wesentlichen Nachteil: Sie weisen Singularitäten auf. Für die Rotationsreihenfolge *Roll–Pitch–Yaw* (XYZ) tritt der kritische Fall in der Nähe von $\theta = \pm 90^\circ$ (Pitch) auf, da zwei Rotationsachsen dabei effektiv zusammenfallen. In diesem Bereich sind Roll und Yaw nicht mehr unabhängig bestimmbar, sodass bereits kleine Änderungen der tatsächlichen Orientierung zu großen oder sprunghaften Änderungen ein-

zelter Euler-Winkel führen können. Dieses Verhalten wird als Gimbal-Lock bezeichnet. (Tamim Asfour 2026)

Um diese Einschränkung zu vermeiden, wird die Orientierung in Filteralgorithmen **Verweis zu 4.2?** intern nicht über Euler-Winkel angegeben, sondern durch Quaternionen beschrieben. Diese Darstellung ist frei von Singularitäten und eignet sich für eine robuste Zustandsführung. Quaternionen sind eine Erweiterung der komplexen Zahlen und lassen sich allgemein in der Form

$$q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k} \quad (2.1)$$

darstellen, wobei $a, b, c, d \in \mathbb{R}$ und $\mathbf{i}, \mathbf{j}, \mathbf{k}$ imaginäre Einheiten sind. Für die Beschreibung von Rotationen im 3-dimensionalen Raum werden in der Praxis Einheitsquaternionen verwendet, die häufig in Vektorform als

$$q = \begin{bmatrix} q_w & q_x & q_y & q_z \end{bmatrix}^T = \begin{bmatrix} q_w & \mathbf{q}_v \end{bmatrix}^T \quad (2.2)$$

geschrieben werden. Dabei bezeichnet q_w den Skalaranteil und

$$\mathbf{q}_v = \begin{bmatrix} q_x & q_y & q_z \end{bmatrix}^T \quad (2.3)$$

den Vektoranteil. Eine Rotation lässt sich mit Quaternionen besonders anschaulich über das Achse-Winkel-Konzept darstellen: Jede räumliche Drehung kann als Drehung um eine bestimmte Achse mit einem bestimmten Winkel beschrieben werden. Quaternionen verpacken diese Information kompakt, indem der Skalarteil den Winkelanteil und der Vektorteil die Drehachse enthält. Dadurch lassen sich Rotationen effizient speichern.

Ein weiterer praktischer Vorteil ist, dass sich aufeinanderfolgende Rotationen einfach verketteten lassen. Wird eine Orientierung nacheinander um mehrere Rotationen verändert, kann die Gesamtrotaion durch Kombination der zugehörigen Quaternionen berechnet werden. Die Hintereinanderausführung von Rotationen entspricht dabei der Quaternion-Multiplikation. Dadurch ist es möglich, Orientierungsänderungen schrittweise zu aktualisieren. **Rotationsmatrix einfügen, falls später verwendet!** (Clemens 2015, S.25-28)

Da die interne Zustandsdarstellung als Quaternionen erfolgt, die Ergebnisse jedoch zur Interpretation als Euler-Winkel angegeben werden, ist eine konsistente Umrechnung zwischen beiden Darstellungen erforderlich. Durch Gleichsetzen der Rotationsbeschreibungen

$$R(\phi, \theta, \psi) = R_x(\phi) R_y(\theta) R_z(\psi) \quad (2.4)$$

und den anschließenden Vergleich der Matrixelemente, ergeben sich folgenden Umrechnungsformeln von Quaternionen zu Euler-Winkeln in Roll-Pitch-Yaw (XYZ) Reihenfolge. Im Folgenden werden die resultierenden Ausdrücke direkt angegeben; eine vollständige Herleitung findet sich in der Literatur. (Diebel 2006, S.24)

$$\phi = \text{atan2}(2(q_w q_x + q_y q_z), 1 - 2(q_x^2 + q_y^2)) \quad (2.5)$$

$$\theta = \arcsin(2(q_w q_y - q_z q_x)) \quad (2.6)$$

$$\psi = \text{atan2}(2(q_w q_z + q_x q_y), 1 - 2(q_y^2 + q_z^2)) \quad (2.7)$$

2.4 Lageschätzung mittels Sensorfusion

Sensorfusion bezeichnet die Kombination von Messdaten mehrerer Sensoren, um die Qualität der Zustandsinformation zu erhöhen. Durch die Nutzung komplementärer Sensoreigenschaften lassen sich Unsicherheiten reduzieren und konsistente Schätzwerte gewinnen. **In dieser Arbeit steht die Orientierungsschätzung (Attitude) einer IMU im Vordergrund.** Da kein Magnetometer eingesetzt wird, basiert die Schätzung auf Gyroskop- und Beschleunigungssignalen. Dadurch können Roll und Pitch langfristig stabilisiert werden, während der Yaw-Winkel ohne externe Referenz grundsätzlich driftbehaftet bleibt.

Eine robuste Orientierungsbestimmung ist mit einem einzelnen Sensor nicht zuverlässig möglich, da die Messgrößen jeweils systematische Einschränkungen besitzen. Das Gyroskop misst die Winkelgeschwindigkeit und erlaubt eine hochdynamische Fortschreibung der Orientierung. Die Messung ist jedoch durch Rauschen beeinflusst und weist typischerweise einen Bias auf.

Unter Bias wird ein systematischer, meist langsam zeit- und temperaturabhängiger Offset verstanden, der dazu führt, dass auch im Stillstand eine von Null abweichende Winkelgeschwindigkeit gemessen wird. Da die Orientierung aus der Winkelgeschwindigkeit durch Integration berechnet wird, führt ein solcher Offset zu einem über die Zeit anwachsenden Fehler und damit zu Drift.

Der Beschleunigungssensor misst die spezifische Kraft. In quasi-statischen Situationen kann daraus die Richtung der Gravitation als Referenz abgeleitet werden, wodurch insbesondere Roll und Pitch langfristig stabilisiert werden. Bei dynamischen Bewegungen überlagern zusätzliche translatorische Beschleunigungen den Gravitationsanteil, sodass die Gravitation nicht mehr eindeutig aus der Messung bestimmbar ist. Aus diesem Grund wird die beschleunigungsbasierte Korrektur in Fusionsverfahren häufig situationsabhängig gewichtet oder bei erkennbar starker Dynamik unterdrückt, um Fehlkorrekturen zu vermeiden.

Die grundlegende Idee der Sensorfusion aus Gyroskop und Beschleunigungssensor besteht in der Kombination beider Eigenschaften. Die Gyroskopintegration liefert eine kurzzeitig genaue Orientierung, driftet jedoch über die Zeit. Die Beschleunigung liefert eine langfristige Referenz, ist aber bei Dynamik verfälschbar. Alle betrachteten Fusionsverfahren folgen dabei dem Prinzip Prädiktion und Korrektur: In der Prädiktion wird die Orientierung aus der Gyroskopmessung fortgeschrieben. In der Korrektur wird die aus der aktuellen Orientierungsschätzung erwartete Gravitationsrichtung mit der Beschleunigungsmessung verglichen. Die daraus resultierende Abweichung wird genutzt, um den Driftanteil der Gyro-Integration zu reduzieren.

?Auf Basis dieser gemeinsamen Struktur werden im Folgenden unterschiedliche Verfahren zur Sensorfusion vorgestellt. Abschnitt 2.5.1 skizziert die prinzipielle Funktionsweise eines (E)KF zur Orientierungsschätzung, ohne eine vollständige Herleitung vorzunehmen, da ein bestehender EKF verwendet wird. Abschnitt 2.5.2 beschreibt komplementärfilterbasierte Ansätze als recheneffiziente Alternative, bevor in Abschnitt 2.5.3 der Mahony-Filter als spezielle Ausprägung des Komplementärfilters eingeordnet und die Auswahl begründet wird. (von Rosenberg 2006, S.25-27)

2.4.1 Erweitertes Kalmanfilter

Der erweiterte Kalmanfilter (KF) ist ein rekursiver Zustandsschätzer für dynamische Systeme, der aus verrauschten Messungen einen möglichst guten Schätzwert des Systemzustands bestimmt. Er basiert auf einem Zustandsraummodell mit Prozessgleichung und Messgleichung und kombiniert dabei Modellwissen mit Sensordaten. Der klassische KF setzt ein lineares Systemmodell sowie lineare Messgleichungen voraus. Der Algorithmus besteht aus zwei wiederkehrenden Schritten: In der Prädiktion wird der Zustand mit dem Systemmodell fortgeschrieben und die zugehörige Unsicherheit (Kovarianz) propagiert. In der Korrektur wird die Vorhersage mithilfe der Messung aktualisiert, wobei der Kalman-Gain die Gewichtung zwischen Modell und Messung bestimmt. Für die IMU-basierte Orientierungsschätzung ist der klassische KF jedoch nur eingeschränkt geeignet, da die zugrunde liegenden Zusammenhänge nichtlinear sind. Die Orientierungskinetik wird über Quaternionen bzw. Rotationsmatrizen beschrieben und ist nicht linear in den Zuständen. Eine direkte Anwendung des linearen KF würde daher die Modellrealität nur unzureichend abbilden oder starke Vereinfachungen erfordern, was zu ungenauen oder instabilen Schätzergebnissen führen kann.

Aus diesem Grund wird in der Praxis ein Erweiterter Kalmanfilter (EKF) eingesetzt. Das EKF überträgt das Prinzip des Kalmanfilters auf nichtlineare Systeme, indem nichtlineare Prozess- und Messmodelle verwendet und in jedem Zeitschritt um den aktuellen Arbeitspunkt linearisiert werden. Die Linearisierung erfolgt über die Jacobi-Matrizen der Prozess- und Messfunktionen. Dadurch bleibt der rekursive Aufbau aus Prädiktion und Korrektur erhalten, während die für die Orientierungsschätzung notwendigen nichtlinearen Rotationsbeziehungen berücksichtigt werden.

ABBILDUNG MA Vgl. Kim(2011) (Kim 2011, S. 146)

Die Abbildung **ref** zeigt den Ablauf des erweiterten Kalmanfilters (EKF) als rekursiven Algorithmus. Das EKF berechnet in jedem Zeitschritt k aus dem vorherigen Schätzwert und der aktuellen Messung einen verbesserten Schätzwert des Zustands. Der Algorithmus lässt sich in vier Teile aufspalten:

0) Initialisierung

Zu Beginn werden ein Startwert \hat{x}_0 sowie die zugehörige Fehlerkovarianz P_0 festgelegt. \hat{x}_0 stellt die erste Zustandsabschätzung dar, P_0 beschreibt die anfängliche Unsicherheit dieser Schätzung.

1) Prädiktion von Zustand und Fehlerkovarianz

Im ersten Schritt wird aus dem vorherigen Schätzwert \hat{x}_{k-1} eine a-priori-Schätzung \hat{x}_k^- gebildet:

$$\hat{x}_k^- = f(\hat{x}_{k-1}). \quad (2.8)$$

Gleichzeitig wird die Unsicherheit dieser Vorhersage fortgepflanzt:

$$P_k^- = A P_{k-1} A^\top + Q. \quad (2.9)$$

Dabei beschreibt A die Linearisierung (Jacobi-Matrix) des Systemmodells um den aktuellen Arbeitspunkt, und Q modelliert Prozessrauschen. P_k^- charakterisiert damit die Unsicherheit der a-priori-Schätzung.

II) Berechnung des Kalman-Gains

Im zweiten Schritt wird der Kalman-Gain K_k berechnet:

$$K_k = P_k^- H^\top \left(H P_k^- H^\top + R \right)^{-1}. \quad (2.10)$$

Der Kalman-Gain legt fest, wie stark die Messung im nächsten Schritt zur Korrektur herangezogen wird. H ist die Linearisierung (Jacobi-Matrix) der Messfunktion, R beschreibt das Messrauschen der Sensorik.

III) Berechnung der korrigierten Schätzung (Update)

Nun wird die a-priori-Schätzung mit der aktuellen Messung z_k korrigiert:

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - h(\hat{x}_k^-)). \quad (2.11)$$

Der Term $z_k - h(\hat{x}_k^-)$ wird als **Innovation** bezeichnet. Er beschreibt die Abweichung zwischen der tatsächlichen Messung z_k und der aus der Vorhersage erwarteten Messung $h(\hat{x}_k^-)$. Diese Abweichung wird mit K_k gewichtet und zur Vorhersage addiert.

IV) Aktualisierung der Fehlerkovarianz

Abschließend wird die Unsicherheit nach dem Update aktualisiert:

$$P_k = P_k^- - K_k H P_k^-. \quad (2.12)$$

Damit liegt für den aktuellen Zeitschritt k sowohl der korrigierte Zustandsvektor \hat{x}_k als auch die zugehörige Fehlerkovarianz P_k vor. Diese Größen dienen im nächsten Zeitschritt wieder als Ausgangspunkt, wodurch der EKF kontinuierlich in einer Schleife arbeitet. (Michaelson 2018)

2.4.2 Komplementärfilter

Der Komplementärfilter ist ein einfaches Verfahren der Sensorfusion, das zwei Messgrößen mit komplementären Fehler- und Frequenzeigenschaften kombiniert, um eine robuste Gesamtschätzung zu erhalten. Aufgrund der geringen algorithmischen Komplexität wird er häufig in Embedded-Systemen eingesetzt, wenn eine Echtzeitschätzung bei begrenzten Rechenressourcen erforderlich ist.

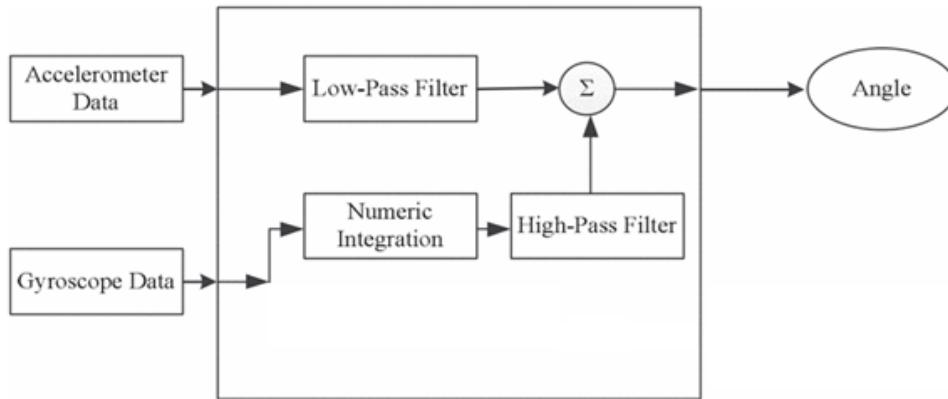


Abb. 2: Blockdiagramm zur Veranschaulichung der Funktionsweise eines Komplementärfilters (Gui, Tang, Mukhopadhyay 2015)

Anhand von Abbildung 2 wird im Folgenden das Funktionsprinzip des Komplementärfilters erläutert. Der Komplementärfilter fusioniert die Messinformationen von Beschleunigungssensor und Gyroskop, indem beide Signalpfade gezielt in unterschiedlichen Frequenzbereichen genutzt werden. Wie in der Abbildung dargestellt, wird aus den Beschleunigungsdaten zunächst eine Neigungsinformation abgeleitet, die anschließend durch einen Tiefpass gefiltert wird. Dieser Pfad trägt damit vor allem die niederfrequenten Anteile der Orientierungsschätzung und liefert insbesondere eine langfristig stabile Referenz über die Schwerkraftrichtung.

Parallel dazu wird die Winkelgeschwindigkeit des Gyroskops numerisch integriert, um eine Winkelschätzung zu erhalten. Diese Schätzung bildet hochfrequente, dynamische Orientierungsänderungen zuverlässig ab, ist jedoch ohne Korrektur driftbehaftet. Durch Hochpassfilterung werden die niederfrequenten Driftanteile unterdrückt, während die hochfrequenten Anteile erhalten bleiben.

Beide Anteile werden im Summationsblock zusammengeführt. Durch die komplementäre Aufteilung in Tiefpassanteil (Beschleunigung) und Hochpassanteil (Gyroskop) entsteht eine Winkelschätzung, die kurzfristig dynamikfähig ist und gleichzeitig langfristig stabilisiert wird. In diskreter Form lässt sich dieses Prinzip exemplarisch durch

$$\theta_k = \alpha \left(\hat{\theta}_{k-1} + \omega_k \Delta t \right) + (1 - \alpha) \theta_{\text{acc},k} \quad (2.13)$$

beschreiben. Dabei bezeichnet $\hat{\theta}_k$ den geschätzten Winkel zum Zeitpunkt k , ω_k die gemessene Winkelgeschwindigkeit, Δt die Abtastzeit und $\theta_{\text{acc},k}$ den aus dem Beschleunigungssignal abgeleiteten Winkel. Der Parameter $\alpha \in (0, 1)$ bestimmt die Aufteilung zwischen beiden Informationsquellen. Große α -Werte gewichten die Gyro-Integration stärker, was eine gute Dynamik ermöglicht, jedoch die Driftkorrektur reduziert. Kleinere α -Werte erhöhen die niederfrequente Stabilisierung durch den Beschleunigungssensor. Dadurch wird die Drift besser unterdrückt, gleichzeitig steigt jedoch die Empfindlichkeit gegenüber dynamischen Zusatzbeschleunigungen. (Gui, Tang, Mukhopadhyay 2015)

Literaturverzeichnis

- Clemens, Joachim (2015):** Entwicklung eines Subsystems „Sensorfusion“ für Navigation und Operatorsupport. In: URL: <https://edocs.tib.eu/files/e01fb16/87093077X.pdf> (Abruf: 12. 01. 2026).
- Diebel, James (2006):** Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors. In: URL: https://pasta.place/Informatik/Robotik_1_%5BHIS%5D/Folien/WS_17-18/01-Mathematische-Grundlagen.pdf (Abruf: 12. 01. 2026).
- Gui, Pengfei; Tang, Liqiong; Mukhopadhyay, Subhas C. (2015):** MEMS based IMU for tilting measurement: Comparison of complementary and kalman filter based data fusion. In: *Proceedings of the 2015 10th IEEE Conference on Industrial Electronics and Applications (ICIEA 2015)*. IEEE, S. 2004–2009. DOI: 10.1109/ICIEA.2015.7334442. URL: <https://ieeexplore.ieee.org/document/7334442> (Abruf: 13. 01. 2026).
- Hoyos-Gutiérrez, Jose; Cardona-Aristizabal, Jaiber; Muñoz-Gutiérrez, Pablo Andrés; Ramirez-Jimenez, Diego (2023):** A Systematic Literature Review on Rapid Control Prototyping Applications. In: *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*. IEEE Xplore Document 10056232. DOI: 10.1109/RITA.2023.3250559. URL: <https://ieeexplore.ieee.org/document/10056232> (Abruf: 12. 01. 2026).
- Iman Bagheri (2026):** Euler Angles (Yaw, Pitch and Roll). URL: https://www.researchgate.net/figure/Euler-Angles-Yaw-Pitch-and-Roll_fig2_342145524 (Abruf: 10. 01. 2026).
- Kim, Phil (2011):** Kalman Filter for Beginners: With MATLAB Examples. In: *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*. (Abruf: 13. 01. 2026).
- Michaelson, Tobias (2018):** Lagebestimmung durch Sensorfusion mittels Kalmanfilter. Masterarbeit. Hamburg, Deutschland: Universität Hamburg. URL: <http://hdl.handle.net/20.500.12738/8460> (Abruf: 13. 01. 2026).
- Tamim Asfour (2026):** Robotik U: Einführung in die Robotik. URL: https://pasta.place/Informatik/Robotik_1_%5BHIS%5D/Folien/WS_17-18/01-Mathematische-Grundlagen.pdf (Abruf: 10. 01. 2026).
- Von Rosenberg, Harald (2006):** Sensorfusion zur Navigation eines Fahrzeugs mit low-cost Inertialsensorik. Public version (PDF). Diplomarbeit. Stuttgart, Deutschland: Universität Stuttgart. URL: <https://www.vrosenberg.de/download/Diplomarbeit%20Harald%20von%20Rosenberg%20%28public%20version%29.pdf> (Abruf: 12. 01. 2026).