

Zieldetektion und Tracking mit einem Echtzeit-MIMO-SONAR-System

Alexej Namenas¹, Thorben Kaak¹, Tim Owe Wisch¹, Gerhard Schmidt¹

¹ *Digitale Signalverarbeitung und Systemtheorie, CAU zu Kiel, E-Mail: {aln, thka, timw, gus@tf.uni-kiel.de}*

Kurzfassung

In diesem Paper wird ein echtzeitfähiges MIMO-SONAR-System beschrieben. Der Fokus liegt dabei auf den Themen Zieldetektion und Tracking. Die Detektion von Zielen ist derart gestaltet, dass sie möglichst unabhängig von den Umgebungsbedingungen robust funktioniert, um die Leistungsfähigkeit des Systems zu maximieren. Die Herausforderungen bei der Detektion und dem Tracking von Zielobjekten sind bei SONAR-Anwendungen vor allem durch das geringe Signal-zu-Rausch-Verhältnis bzw. dessen zeitlicher Variabilität begründet. In diesem Paper werden die Systemanforderungen und die daraus resultierende Systemstruktur mit den darin enthaltenen Modulen der Signalverarbeitung erklärt. Eine besondere Herausforderung des Systems ist der Ressourceneinsatz sowie die Verwaltung der Ressourcen (Speicher und Rechenleistung), die im Hinblick auf sich ändernde Umgebungsbedingungen zu steuern sind. In diesem Zusammenhang soll das System dabei helfen, Algorithmen und Signalverarbeitungsverfahren, die ihre Anwendung in dem MIMO-SONAR-System finden, anhand von echten SONAR-Daten zu evaluieren. Damit handelt es sich um ein funktionsfähiges System, das mit realen SONAR-Daten arbeitet und welches durch weitere Komponenten, wie der Positionsbestimmung und dem Einsatz auf dem Wasser anhand von wohldefinierten Szenarien evaluiert werden kann. Ebenso werden das verwendete SONAR-System und dessen Spezifikationen, sowie die Möglichkeiten und Herausforderungen, die damit verbunden sind, vorgestellt.

Einleitung

Wie schon in der Kurzfassung spezifiziert, liegt der Fokus des Papers auf der Konzipierung und Vorstellung eines gesamten MIMO-SONAR-Systems. Das Paper lässt sich in drei Abschnitte unterteilen. Zuerst soll es einen Einblick in das Konzept des von der Arbeitsgruppe erarbeiteten MIMO-SONAR-Systems geben. Das hier vorgestellte Konzept beschränkt sich ausschließlich auf die eigentliche Verarbeitung der empfangenen SONAR-Signale. Die dahinter stehende Hardware wird nicht vorgestellt. In einem vorigen DAGA Beitrag [1] wurde die Idee des kognitiven Systems vorgestellt, das ein Teil dieser Anwendung darstellt. In diesem Fall handelte es sich um die Hauptkontrollinheit (*Main Control Unit*), die in Abbildung 1 als blau umrandeter Verarbeitungsblock dargestellt ist. Als Ergänzung zu dem gesamten System werden zwei Module vorgestellt, das Detektion-Modul und das Tracking-Modul. Diese sind in Abbildung 1 rot umrandet dargestellt. Sie stehen in direkter Nachbarschaft zu der *Main Control Unit*, kurz MCU, und versorgen diese mit Informationen zu der aktuellen Detektions-Lage, sowie zu

den generierten Tracks, auf Basis welcher das SONAR-System angepasst und die Ressourcen gemanagt werden. Das Detektion-Modul, so wie es derzeit vorliegt, wurde bereits implementiert. Eine genauere Beschreibung ist in [3] zu finden und wird hier nur oberflächlich behandelt. Dem Tracking-Modul, das dem Detektion-Modul nachgeschaltet ist, obliegt die Aufgabe die große Menge der Kontakte (Messwerte) zu analysieren. Bei der Analyse der Daten muss darüber entschieden werden, ob die identifizierten Kontakte einem existierenden Track hinzugefügt werden sollen, ob ein neuer Track aufgemacht wird und ob ein Track verworfen werden soll. Aufgrund einer ständig wachsenden Datenmenge muss ein Kompromiss zwischen einer verträglichen Auslastung und der Tracking-Genauigkeit geschaffen werden, um die Echtzeitfähigkeit zu gewährleisten.

Konzept des MIMO-SONAR-Systems

Die in Abbildung 1 dargestellte Struktur des MIMO-SONAR-Systems veranschaulicht den Verarbeitungsflussgraphen und stellt somit eine zukünftige Vision eines echtzeitfähigen Rahmenwerks dar. Das System lässt sich grob in vier Verarbeitungsblöcke unterteilen. Der erste Block stellt die empfangsseitige Verarbeitungskette dar. Er besteht aus den Hauptmodulen Vorverarbeitung/Analyse-Filterbank, dem Empfangsbeamformer und dem Mehrkanal-Korrelator. Der zweite Block ist die sendeseitige Verarbeitungskette, die sowohl konventionell SIMO- als auch MIMO-fähig ist. In diesem Block sind die Module Signalgenerierung, Synthese-Filterbank als auch Normierung/Kalibrierung zu finden. Das Modul der Signalgenerierung stellt gleichzeitig die Signale für das empfangsseitige Matched-Filter bereit. Der dritte Block besteht aus zwei Modulen. Das Detektions-Modul nutzt die Korrelator-Daten in Form von Range-Matrizen zur Detektion von Zielobjekten. Das letzte Modul verwendet die Detektions-Daten für die Generierung von Tracks. Die Range-Matrix, die darin detektierten Zielobjekte und daraus generierten Tracks werden dem vierten Block zur Verfügung gestellt. Dieser Block stellt die MCU [1] dar. Sie soll anhand von den eintreffenden Daten die drei erwähnten Verarbeitungsblöcke steuern. Die Steuerung soll vorerst nach fest definierten Regeln ablaufen.

Detektion von Zielobjekten

Die Detektion von Zielobjekten in SONAR-Anwendungen unterscheidet sich nur leicht von den Anwendungen im Radar-Bereich. Im SONAR-Bereich werden die Detektions-Algorithmen mit einem relativ schlechten Signal-zu-Rausch-Verhältnis konfrontiert. Ein

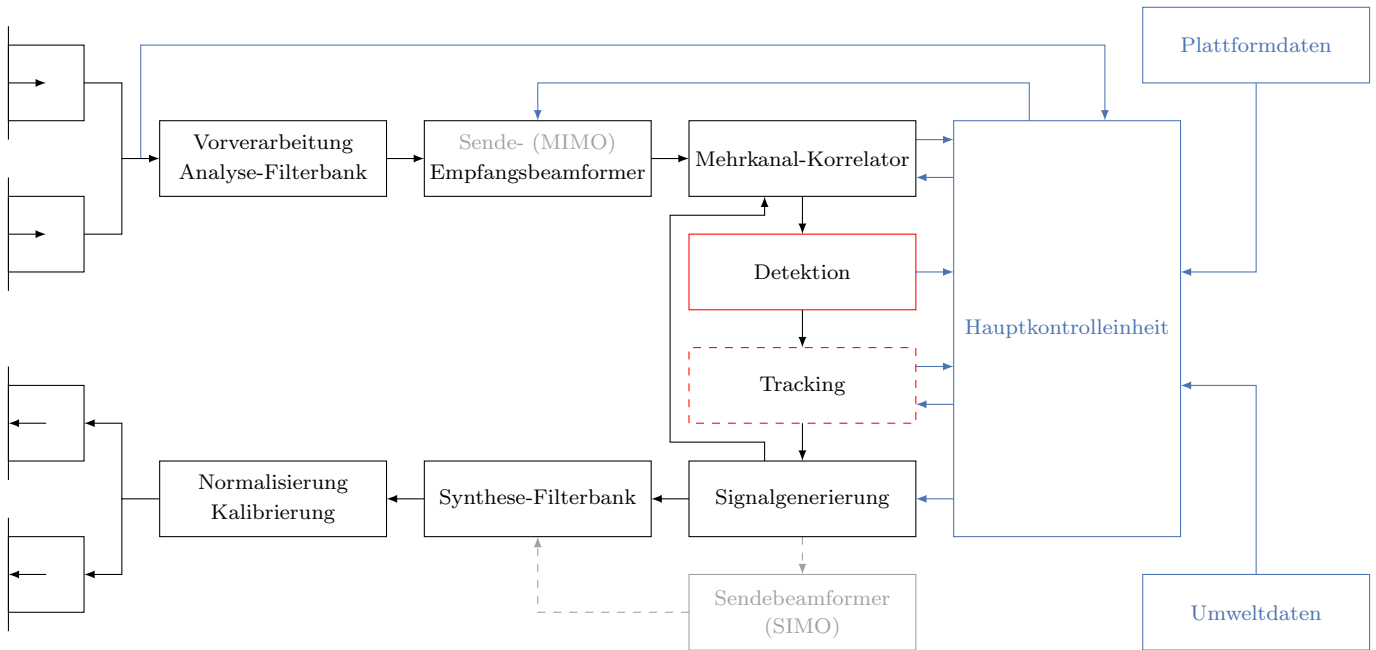


Abbildung 1: Struktur des MIMO-SONAR-Systems.

breites Spektrum von Detektionsverfahren bietet unter anderem vielseitige Möglichkeiten auf unterschiedliche Szenarien-Abläufe zu reagieren. Die wohl bekanntesten Detektions-Algorithmen beruhen auf der *Constant False Alarm Rate*, kurz CFAR [5]. Die Unterschiede liegen

dienen der Bestimmung des adaptiven Schwellwertes. Wie in der Abbildung 2 dargestellt, lassen sich die Schwellwerte entweder mit dem größten Mittelwert adaptieren, entsprechend dem *Greatest Of CA-CFAR*, kurz CAGO-CFAR [7], oder mit dem kleinsten Mittelwert adaptieren, entsprechend dem *Smallest Of CA-CFAR*, kurz CASO-CFAR [7]. Ein weiteres, hier nicht dargestelltes Verfahren, bedient sich einer zuvor sortierten Stichprobe. Der Algorithmus nennt sich *Ordered Statistics CFAR*, kurz OS-CFAR [8]. Er wendet unter anderem das aus der Statistik bekannte Vorgehen der Bereinigung einer Stichprobe, mit dem Ziel der Reduktion des Einflusses der sogenannten Ausreißer, für ein repräsentativeres Ergebnis. Die Erweiterung der Algorithmen auf eine weitere, zeitliche Komponente ermöglicht eine Verbesserung der Detektionsrate bei einer gleichbleibenden Falschalarmrate, was dem Informationszuwachs zu verdanken ist. Gleichzeitig ermöglicht die zeitliche Betrachtung das Aussortieren von zumeist zeitlich- und räumlich-variantem Clutter. Dieses Verfahren wurde im Rahmen einer Masterarbeit entwickelt und in das Echtzeitrasterwerk integriert. Die Rede ist vom *Time-reinforced Variability Index CFAR* [3, 9], kurz TRVI-CFAR. In diesem Paper wird nicht auf das Verfahren eingegangen, es sei jedoch bemerkt, dass es sich hierbei um eine Fusion von mehreren Verfahren handelt, die eine Verbesserung der Detektionsleistung bei gleichbleibender Falschalarmrate bewirkt.

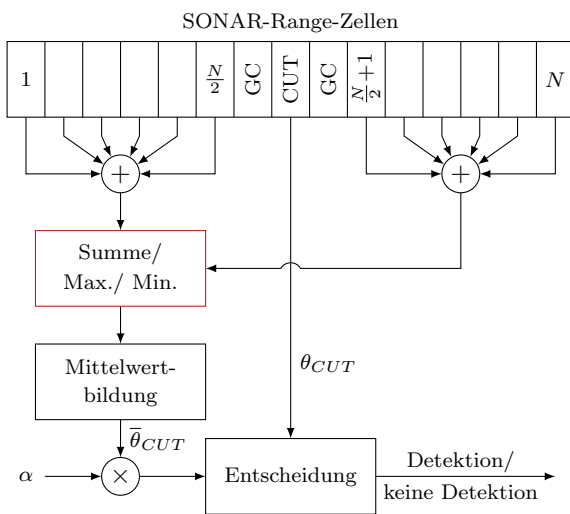


Abbildung 2: Das Prinzip des CA-CFAR Algorithmus [4]. Die oberen SONAR-Range-Zellen stellen einen ausgewählten Bereich der gesamten SONAR-Range-Matrix dar. Im rot umrandeten Verarbeitungsblock wird das entsprechende CFAR-Verfahren, wie CA, CAGO oder CASO ausgewählt.

in der Definition der Stichproben für die anschließende adaptive Anpassung der Schwellwerte. Es lassen sich zeitliche und räumliche CFAR-Algorithmen als auch Kombination dieser determinieren. In Abbildung 2 ist beispielhaft das CA-CFAR-Verfahren [6] für eine Stichprobengröße N dargestellt. Die in der Mitte liegenden Zellen CUT (*Cell Under Test*) und GC (*Guard Cell*) sind kein Teil der Stichprobe. Die restlichen Zellen

Tracking von Zielobjekten

Bei der Objektverfolgung geht es um die dynamische Schätzung von Zuständen \mathbf{x}_k und der Anzahl von Objekten. Der Zustand eines Objektes zum Zeitpunkt t_k kann anhand eines Systemmodells beschrieben werden

$$\mathbf{x}_k = f_k(\mathbf{x}_{k-1}) + \mathbf{w}_k, \quad (1)$$

mit

$$\mathbf{x}_k = [x_k \quad v_{x,k} \quad y_k \quad v_{y,k}]^T. \quad (2)$$

Die Zustandsabhängige Funktion f_k steht für die Systemfunktion oder Bewegungsgleichung. Diese kann im ersten Schritt der Implementierung als zeitinvariant angenommen werden. Die Variable \mathbf{w}_k beschreibt das additive Prozessrauschen und soll die Ungenauigkeiten bei der Approximation der Systemfunktion simulieren. Neben dem Systemmodell wird die Messgleichung

$$\mathbf{z}_k = h_k(\mathbf{x}_k) + \mathbf{v}_k \quad (3)$$

mit

$$\mathbf{z}_k = [x_k \quad y_k \quad \dot{r}_k]^T. \quad (4)$$

definiert. Die Variable \mathbf{v}_k steht für das Messrauschen und beschreibt Ungenauigkeiten der zustandsabhängigen Messfunktion, die ebenfalls nur approximiert angegeben werden kann. Die in der Abbildung 3 dargestellte Struktur des Trackers stellt das anvisierte Zustandsschätzungsmodell dar und soll im Weiteren erläutert werden.

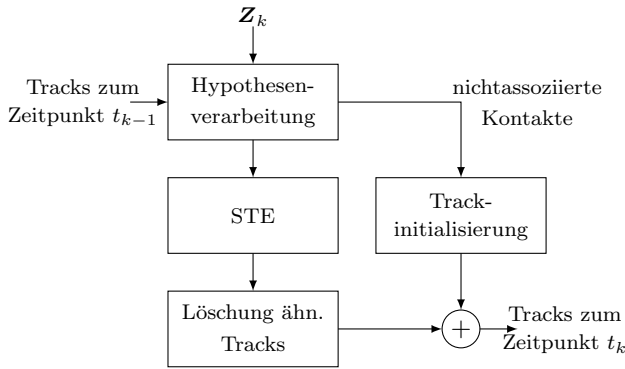


Abbildung 3: Struktur der Trackverarbeitung [4].

Linearer Kalman-Filter

Die Schätzung der Zustände bedient sich eines Linearen Kalman-Filters (LKF). Die Messwerte \mathbf{z}_k zusammen mit den Kovarianzmatrizen \mathbf{Q}_k und \mathbf{R}_k des Prozess- bzw. Messrauschens dienen der Zustandsschätzung $\hat{\mathbf{x}}_k$, die in zwei Schritten vollzogen wird. Im ersten Schritt erfolgt eine Prädiktion:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1}, \\ \hat{\mathbf{P}}_{k|k-1} &= \mathbf{F}\hat{\mathbf{P}}_{k-1|k-1}\mathbf{F}^T + \mathbf{Q}_k. \end{aligned} \quad (5)$$

Im zweiten Schritt findet die eigentliche Schätzung des Zustandes statt:

$$\begin{aligned} \boldsymbol{\gamma}_k &= \mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}, \\ \mathbf{S}_k &= \mathbf{H}\hat{\mathbf{P}}_{k|k-1}\mathbf{H}^T + \mathbf{R}_k, \\ \mathbf{K}_k &= \hat{\mathbf{P}}_{k|k-1}\mathbf{H}^T\mathbf{S}_k^{-1}, \\ \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\boldsymbol{\gamma}_k, \\ \hat{\mathbf{P}}_{k|k} &= \hat{\mathbf{P}}_{k|k-1} - \mathbf{K}_k\mathbf{H}\hat{\mathbf{P}}_{k|k-1}. \end{aligned} \quad (6)$$

Das LKF bildet somit zwei Untermodule des Hypothesenverarbeitungs-Moduls.

Hypothesenverarbeitung

Das Hypothesenverarbeitungs-Modul, siehe Abbildung 4, dient der Zuordnung der Messwerte \mathbf{z}_k zu den schon existierenden Tracks mit vorheriger Auswahl (*Gating*) von wahrscheinlichsten Messwerten um die geschätzten Messwerte $\hat{\mathbf{x}}_k$. Nichtassoziierte Kontakte werden nicht weiter verarbeitet.

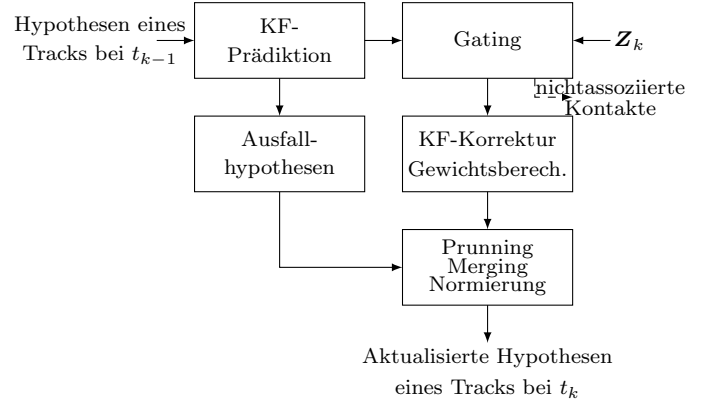


Abbildung 4: Struktur der Hypothesenverarbeitung eines Tracks [4].

Track Initialisierung

In dem Hypothesenverarbeitungs-Modul können Kontakte auftreten, die keinem Track zugeordnet werden, die sogenannten nichtassoziierten Kontakte. Diese können dennoch von Zielobjekten stammen und sollten nicht verworfen werden. Aus diesem Grund werden diese an das Modul Track Initialisierung übergeben. Alle nicht assoziierten Kontakte stellen somit den Anfang eines Tracks dar.

Sequential Track Extraction (STE)

Bei der Initialisierung von Tracks besteht die Gefahr von Falschinitialisierungen aufgrund von Fehldetektionen. Um den Tracks eine messbare Bewertung bezüglich ihrer Glaubwürdigkeit zu geben, soll eine Methode gefunden werden diese messbar zu machen. Ein Verfahren der Bewertung ist in [10] beschrieben und soll seine Anwendung im Tracker finden. Das Verfahren nennt sich *Sequential Track Extraction*, kurz STE, und bedient sich der Berechnung der *Likelihood Rate*:

$$LR(Z_{1:k}^l) = \frac{p(Z_{1:k}^l|H_1)}{p(Z_{1:k}^l|H_0)}. \quad (7)$$

Es werden zwei Hypothesen aufgestellt:

- H_0 : $Z_{1:k}^l$ enthält keine Zielmessungen,
- H_1 : $Z_{1:k}^l$ enthält mindestens eine Zielmessung.

Ist der Wert $LR(Z_{1:k}^l)$ berechnet, so können insgesamt drei Fälle auftreten:

- $LR(Z_{1:k}^l) \geq A$: Hypothese H_1 wird angenommen,
- $LR(Z_{1:k}^l) \leq B$: Hypothese H_0 wird angenommen,

- $A > LR(Z_{1,k}^l) > B$: es wird weder die Hypothese H_0 noch die Hypothese H_1 angenommen und der Track wird bis zum Eintreten eines der ersten beiden Fälle beibehalten.

Die beiden Grenzen A und B werden durch die Wahl der Wahrscheinlichkeiten P_0 und P_1 festgelegt

$$\begin{aligned} A &= \frac{P_1}{P_0}, \\ B &= \frac{1 - P_1}{1 - P_0}, \end{aligned} \quad (8)$$

welche wiederum zuvor festzulegen sind.

Löschung ähnlicher Tracks

Bei der Schätzung von Tracks kann durchaus der Fall auftreten, dass zwei oder mehrere Tracks nah beieinander verlaufen bzw. aufeinander zulaufen und somit kaum zu unterscheiden sind. In diesem Fall sind die Tracks nicht mehr auseinanderzuhalten und sind somit doppelt vertreten. Um einen der beiden Tracks zu entfernen, muss der vektorielle Abstand der Schätzwerte unter einen Schwellwert L_T fallen. Zur Berechnung der Entfernung wird die Mahalanobis-Distanz

$$\delta_{Mh}^2(l, n) = (\mathbf{x}_k^l - \mathbf{x}_k^n)^T (\mathbf{P}_k^l + \mathbf{P}_k^n)^{-1} (\mathbf{x}_k^l - \mathbf{x}_k^n) \quad (9)$$

verwendet. Liegt diese unter dem Schwellwert, so ist einer der beiden Tracks zu verwerfen.

Fazit und Ausblick

In diesem Paper wurde das Konzept des MIMO-SONAR-Systems mit Schwerpunkt Detektion und Tracking zusammenfassend vorgestellt. Hauptaugenmerk wurde auf das Thema Tracking von Zielobjekten gelegt, das im Weiteren verfolgt werden soll. Das vorgestellte Trackingmodell beruht auf der Dissertationsarbeit von Viet Duc Nguyen [4] und soll in das Echtzeit-Rahmenwerk des SONAR-Systems integriert werden.

Eine Herausforderung in diesem Zusammenhang stellt die Aufrechterhaltung der Echtzeitfähigkeit des gesamten Systems dar. Abgesehen von der ausstehenden Implementierung des Moduls, soll eine Evaluierung des gesamten Systems anhand von echten SONAR-Daten durchgeführt werden. Die Evaluierung eines Systems anhand von echten Daten erfordert zusätzliche Kenntnisse des Messablaufs, was in Vergleich mit einer Simulation nicht als bekannt und konstant angenommen werden kann. Dazu zählen unter anderem, das zeitvariante Wasserschallprofil, das periodisch zu messen ist, die genaue Position des Zielobjektes während der Messung, sowie die Position und Ausrichtung der Hydrophone, die aufgrund von Wetterverhältnissen variabel sein können. Anhand der Evaluationsergebnisse können im Nachhinein Schwachstellen des Systems erkannt und Verbesserungsalternativen erarbeitet werden.

Literatur

- [1] Thorben Kaak und G. Schmidt: *An Introduction to Real-time Cognitive SONAR Systems Utilizing Novel MIMO Approaches*, DAGA, 2017, 48-51.
- [2] Anton Namenas, T. Kaak und G. Schmidt: *Real-time Simulation of Underwater Acoustic Channels*, DAGA, 2017, 168-171.
- [3] Thorben Kaak, K. Rebbe und G. Schmidt: *Verbesserung der Detektionsleistung durch den TRVI-CFAR Algorithmus*, DAGA, 2018.
- [4] Viet Duc Nguyen: *Einfluss von Gating-Algorithmen in der Objektverfolgung am Beispiel des Multi-Hypothesentracking*, Aachen, Shaker Verlag, 2015.
- [5] Albrecht K. Ludloff: *Praxiswissen Radar und Radar-signalverarbeitung*, 3. illustrierte Ausgabe, Springer-Verlag, 2013.
- [6] Ramon Nitzberg: *Clutter Map CFAR Analysis*, IEEE Trans. on AES., vol. 22, No.4, July 1986.
- [7] Z. Ebrahimian, *Complex spatial/temporal CFAR*, IEEE/ACES International Conference on Wireless Communications and Applied Computational Electromagnetics, 2005., 2005, pp. 1045-1049.
- [8] A.R. Elias-Fuste/M.G.G. de Mercado/E. de los Reyes Davo: *Analysis of some modified ordered statistic CFAR OSGO and OSSO CFAR*, IEEE Transactions on Aerospace and Electronic Systems, 26 (Jan. 1990), 197-202.
- [9] Amit Kumar Verma: *Variability Index Constant False Alarm Rate (VI-CFAR) for Sonar Target Detection*, IEEE-International Conference on Signal processing, Communications and Networking Madras Institute of Technology, Anna University Chennai India, Jan 4-6, 2008.
- [10] G. Van Keuk: *Sequential Track Extraction*, IEEE Transaction on Aerospace and Electronic Systems, 34(4): 1135 - 1148, Oktober 1998.