

함정 전투체계 표적 융합 정확도 향상을 위한 알고리즘 연구

A Study on Multi Sensor Track Fusion Algorithm for Naval Combat System

정영란*

Jung, Young Ran

ABSTRACT

It is very important for the combat system to process extensive data exactly at short time for the better situation awareness compared with the threats in these days. This paper suggests to add radial velocity on the decision factor of sensor data fusion in the existing algorithm for the accuracy enhancement of the sensor data fusion in the combat system.

주요기술용어(주제어) : Combat System(함정 전투체계), Local Sensor Track Fusion(센서 표적 융합), Position Difference(위치 오차), Radial Velocity(시선 속도)

1. 머리말

함정 전투체계는 함정에 탑재된 센서 및 무장 체계, 전술 데이터 통신으로부터 정보를 수신 받아 전장 상황을 감시하고, 함정 지휘관의 지휘 결심을 돕기 위해 전술 정보를 지휘관 및 운용자에게 전시하고, 탐지 및 식별된 표적에 대해 위협 평가를 수행하여, 위협에 대해 전술 대응을 하는 전투 함정의 두뇌와 같은 역할을 하는 체계이다^[1,2].

나날이 발전하고 있는 대공 위협에 대해, 함정 전투체계는 짧은 반응 시간을 통한 다양한 전술 대응 방법으로 자함의 생존율을 증가시키고 있다. 위협에 대한 짧은 반응 시간은 위협을 먼저 탐지하고 식별하여 상황을 감시할 수 있는 정확한 정보 획득 능력이 관건이다. 이를 위해 각 센서들은 탐지 능력을 계속

향상시키고 있으며, 함정에서는 탑재하는 센서의 종류를 다양화하여 정보 획득 능력을 강화하고 있다.

함정의 센서 다양화로 인해 각종 센서들로부터 전투체계로 수신되는 정보가 지속적으로 증가하고 있으며, 함정 전투체계에서는 이들 정보들을 짧은 시간에 정확하게 처리하는 능력을 중요시 하고 있다. 또한, 이러한 처리 능력을 향상시키기 위해 데이터 처리 알고리즘들에 대한 기술 개발 및 기술 축적 노력이 지속적으로 이루어지고 있다.

함정 전투체계의 주요 데이터 처리 알고리즘 중 하나인 전투체계 데이터 융합은 함정에 탑재된 센서들이나 전술 데이터 통신으로부터 수신된 표적 정보들을 처리 및 통합하여 전술 표적을 생성하는 것이다. 일반적으로 함정 전투체계에서 여러 센서 표적 정보들을 표적 융합하는데 있어, 표적 융합의 정확성을 위해 위치 오차뿐만 아니라, 센서 표적 번호, 식별 정보와 같은 정보들을 융합의 표적 동일성 판단 조건으로 사용하고 있다^[6,7]. 하지만, 표적 융합의 결정적 판단 조건은 위치 오차로써, 위치 오차가 작은 표적

† 2007년 3월 5일 접수~2007년 4월 27일 게재승인

* 국방과학연구소(ADD)

주저자 이메일 : muddy02@add.re.kr

들이 여러 개 존재할 경우 자칫 잘못된 표적 융합을 초래할 수도 있다.

이에 본 논문에서는 표적 융합 정확도를 향상시키기 위해, 시선 속도 성분을 융합의 판단 조건으로 추가 고려하는 표적 융합 알고리즘을 제안하고자 한다.

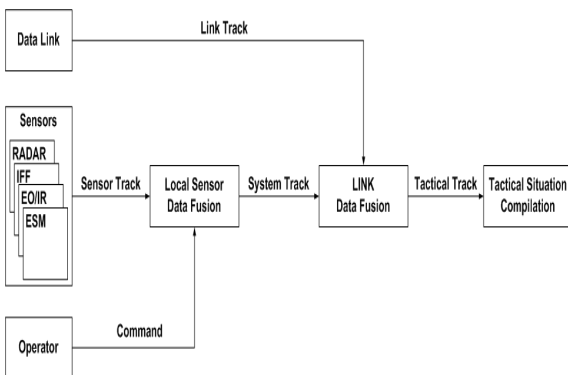
본 논문의 2장에서는 기존 전투체계에 적용된 표적 융합 방법을 분석하고, 3장에서는 표적 융합 정확도를 향상시키기 위해 시선 속도 성분을 추가로 고려한 표적 융합 방법을 제안하고, 이를 실제 적용하기 위한 해석 방법을 제시한다. 그리고 4장에서는 기존 모델과 제안 모델에 대한 성능평가 결과를 제시하며, 5장에서 결론을 맺는다.

2. 전투체계 데이터 융합

전투체계의 데이터 융합은 크게 다음과 같이 구분할 수 있다^[1,2,4,5].

- 센서 표적 융합
(Local Sensor Track Fusion)
- 전술 데이터 통신 표적 융합
(LINK Track Fusion)
- 전술 상황 분석 - 표적 식별 및 위협 평가
(Tactical Situation Compilation)

센서 표적 융합은 자함에 탑재된 다중 센서로부터 수신 받은 센서 표적 정보들을 이용하여 전투체계 시스템에서 사용할 시스템 표적을 생성하는 처리 과정이다.



[그림 1] 전투체계 데이터 융합 구조

전술 데이터 통신 표적 융합은 전술 데이터 통신을 통해 자함으로 수신된 상위 제대 또는 다른 함정의 표적 정보들과 자함의 시스템 표적 정보들을 사용하여 최종적으로 운용자에게 전신될 전술 표적을 생성하는 처리 과정이다. 전술 데이터 통신을 통해 수신되는 상위 제대나 다른 함정의 표적 정보들은 지휘관의 명령과 같은 전술 정보가 포함되어 있기 때문에 센서에서 바로 수신되는 표적들과 다르게 구분하여 처리한다.

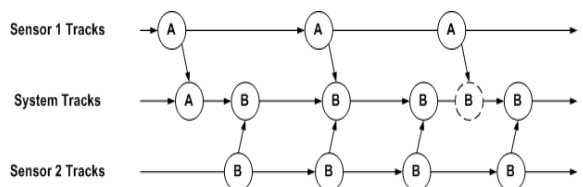
전술 상황 분석 - 표적 식별 및 위협 평가는 SEI (Specific Emission Identifier)와 같이 데이터 베이스화 되어 있는 정보들이나 첩보 정보들을 사용하여 센서 표적 융합과 전술 데이터 통신 표적 융합을 통해 생성된 전술 표적에 대해 위협을 식별하고 평가하는 처리 과정이다.

이들 전투체계 데이터 융합 중에서도 다중 센서로부터 수신 받은 센서 표적 정보들을 이용하여 전투체계의 시스템 표적을 만드는 센서 표적 융합 분야가 본 논문에서 고려하는 분야이다.

일반적으로 데이터 융합 분야에서는 정보 처리를 위한 여러 가지 융합 방법이 존재하지만^[3], 많은 양의 정보를 짧은 시간에 처리해야 하는 전투체계에서는 시스템의 계산량이나 통신 부하량을 줄일 수 있는 분산식 융합 방식을 많이 사용하고 있다.

가. 센서 표적 융합

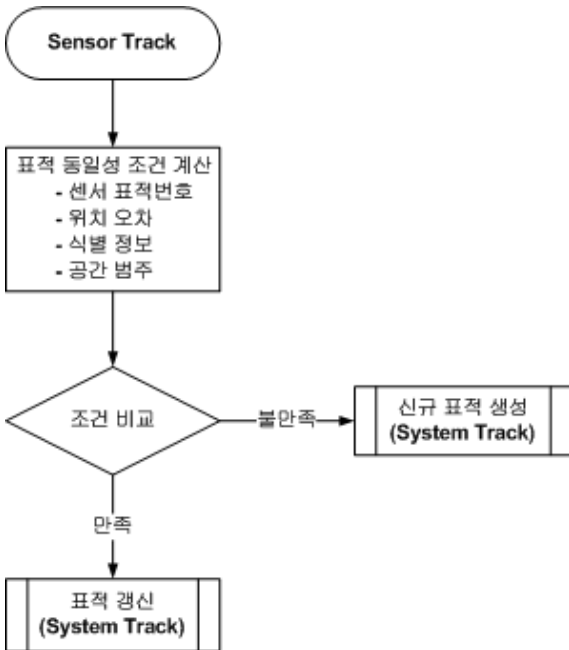
분산식 데이터 융합 방식을 사용한 센서 표적 융합에서는 다중 센서에서 필터링된 센서 표적 정보들이 전투체계 시스템으로 보내지면, 시스템에서는 기존에 존재하는 시스템 표적과의 동일성 여부를 판단하기 위해, 그림 2와 같이 수신된 센서 표적의 시간으로 시스템 표적을 추정(Extrapolation)한 후, 표적 동일성 검사를 수행한다^[6].



[그림 2] 기존전투체계 센서데이터 융합 구조

표적 동일성 검사에서는 크게 다음과 같은 요소들이 고려된다.

- 센서 표적 번호
- 표적 식별 정보(아군, 적군 등)
- 위치 공간 범주(대공, 대함 등)
- 위치 오차(거리 및 고도 고려)



[그림 3] 표적 융합 구조

그림 3과 같이 표적 동일성 조건들의 만족 여부에 따라 신규 시스템 표적 생성이나 표적 갱신으로 구분하여 표적 융합을 수행한다. 이러한 센서 표적 융합 과정은 각 센서 표적들이 수신될 때마다 반복된다.

나. 표적 동일성 검사 - 위치 오차

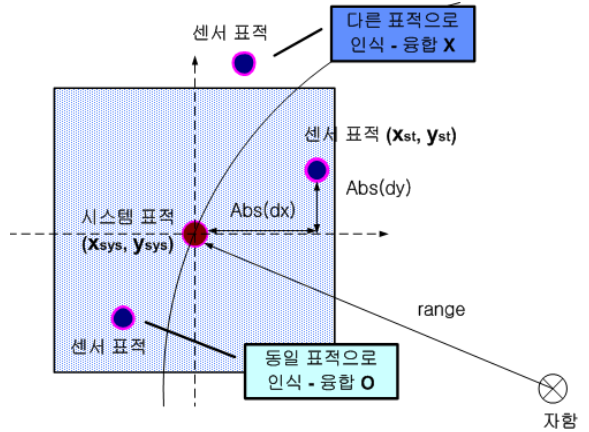
표적의 동일성 검사 조건 중에서 표적 융합 정확도에 가장 영향을 많이 주는 것은 위치오차 조건이다.

다음의 수식이 기존 전투체계에서 고려하는 표적의 위치 오차 조건이다^[6,7].

- 위치오차(거리 및 고도 고려)

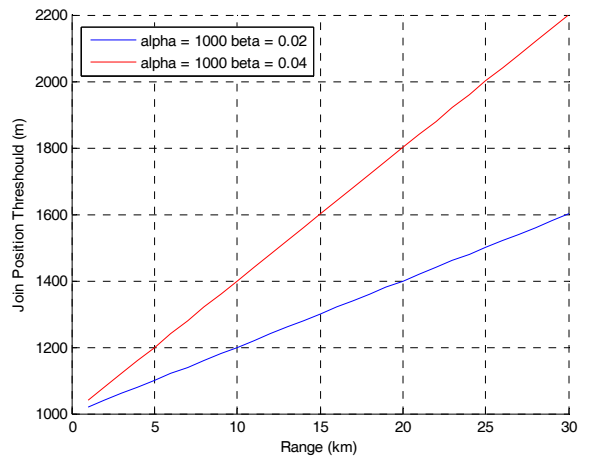
$$Abs(dx) < \alpha + \beta * Abs(range)$$

$$Abs(dy) < \alpha + \beta * Abs(range)$$



[그림 4] 표적 융합 위치 오차 조건

여기서, $dx = x_{sys} - x_{st}$, $dy = y_{sys} - y_{st}$ 로 시스템 표적과 센서 표적의 위치 오차를, range는 자함 기준의 시스템 표적 거리를 의미하고, α 는 시스템 표적과 센서 표적을 표적 융합 가능하게 하는 최대 허용 위치 오차 상수이며, β 는 자함과 표적의 상대적인 거리를 고려하기 위한 상수이다.



[그림 5] 거리에 따른 최대 허용 위치 오차 상수

그림 5에서 보는 바와 같이 자함과 표적의 상대적인 거리에 따라 표적 융합의 위치 오차 조건이 달라질 수 있다.

표적 융합의 위치 오차 조건 값을 센서의 탐지 정확도, 필터링된 센서 표적 정보의 정보 손실 및

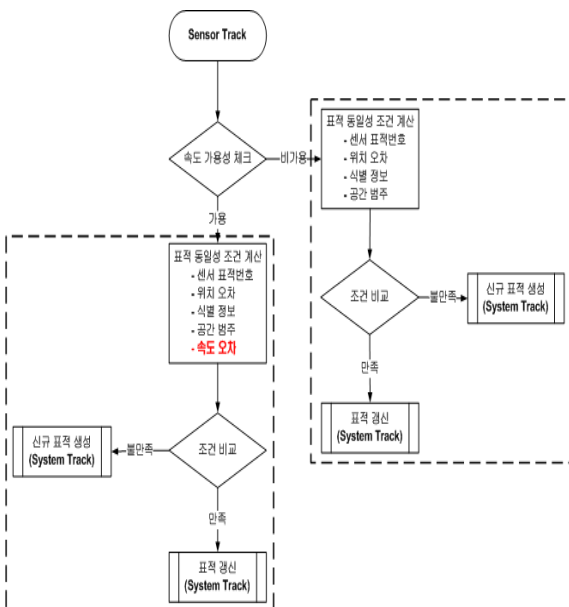
시스템의 데이터 프로세싱 오차를 고려한 최적의 값으로 설정한다면 위치 오차 조건만으로도 정확한 표적 융합을 수행할 수 있다. 하지만, 함정에 탑재되는 센서 종류가 다양하고, 이들의 탐지 정확도 및 필터링된 센서 표적 정보의 정보 손실 정도가 모두 다르기 때문에 이와 같은 정보들을 모두 고려한 최적의 위치 오차 조건 값을 설정하는 것은 매우 어렵다. 반대로 이와 같은 정보들을 고려하지 않을 경우, 동일 표적도 다른 표적으로 인식하게 되는 결과를 초래한다.

이에 본 논문에서는 표적 융합의 정확도를 향상시키기 위해 표적 융합의 표적 동일성 판단 조건으로 위치 오차, 센서 표적 번호, 표적 식별 정보, 위치 공간 범주 외에 속도 성분을 추가로 고려하고자 한다.

3. 전투체계를 위한 표적융합

가. 제안 알고리즘

속도 성분의 경우, 탑재하는 센서의 종류 및 상황에 따라 수신 여부가 달라지기 때문에 먼저 속도 성분의 가용성 여부 검사가 필요하다.



[그림 6] 속도 성분을 고려한 표적 융합 구조

그림 6과 같이 속도 성분이 가용할 경우, 기존 체계의 표적 동일성 조건 외에 속도 오차 조건을 추가로 만족하는 경우에 한해서만 표적 융합의 표적 갱신 처리 과정을 수행하도록 하고, 속도 성분이 가용하지 않을 경우에는 기존 체계의 표적 동일성 조건을 만족할 경우 표적 융합의 표적 갱신 처리 과정을 수행하도록 한다.

속도 성분으로 시선 속도(Radial Velocity)를 고려한다. 시선 속도는 물체의 실제 이동 속도를 나타내는 것이 아니라, 자함으로부터 멀어지거나 다가오는 물체의 속도 성분을 의미한다. 시선 속도는 자함 방향으로 멀어지거나 다가오는 운동 성분만을 표현하기 때문에 각각의 속도 성분 값보다 훨씬 변화에 덜 민감하고, 표적 융합을 위한 표적 동일성 검사 조건을 간단하게 설정할 수 있어 표적의 동일성 여부를 판단할 수 있는 능력은 향상되면서, 조건 추가로 야기되는 프로세서 부하를 최소화 할 수 있다.

다음 두 수식은 본 논문에서 추가로 고려하는 표적 융합을 위한 표적의 동일성 검사 속도 조건들이다.

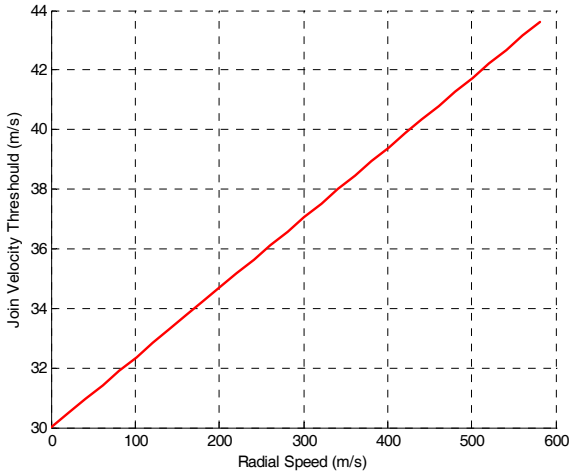
$$v_{rs} * v_{st} \geq 0 \tag{1}$$

여기서, v_{rs} 는 자함 기준의 시스템 표적 시선 속도 성분이며, v_{st} 는 자함 기준의 센서 표적 시선 속도 성분으로, (1)의 식은 자함을 기준으로 같은 방향 성분을 가지는 시선 속도만을 고려하기 위함이다.

$$Abs(dv_r) < \gamma + \delta * Abs(v_{rs}) \tag{2}$$

여기서, dv_r 은 시스템 표적과 센서 표적 사이의 상대적인 시선 속도 차이이며, γ 는 시스템 표적과 센서 표적을 표적 융합 가능하게 하는 최대 허용 속도 오차 상수이며, δ 는 자함 기준의 시스템 표적 시선 속도를 고려하기 위한 상수로, (2)의 식은 시스템 표적을 기준으로 시선 속도 차가 일정 크기 내인 센서 표적만을 고려하기 위함이다.

그림 7에서 보는 바와 같이 자함에 접근하는 표적의 시선 속도에 따라 표적 융합의 속도 조건이 달라질 수 있으며, γ 라는 상수의 조정으로 속도 오차 조건이 변경 가능하다.



[그림 7] 시선 속도에 따른 최대 허용 속도 오차 상수

나. 제안 알고리즘 해석

속도 성분으로 고려하는 시선 속도는 다음과 같이 정의한다.

$$v_r = \frac{dr}{dt}, \quad r = [x^2 + y^2 + z^2]^{\frac{1}{2}} \quad (3)$$

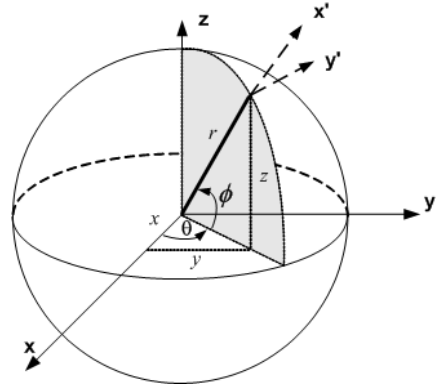
여기서, x, y, z 값은 자함과 표적간의 상대 위치 (Relative Position) 성분이다.

일반적으로 전투체계에서 운용자에게 전시되는 표적 정보들은 거리(r), 방위(θ) 값을 사용하고 있지만, 처리단에서 계산되어지거나 센서로부터 수신되는 표적 정보들은 x, y, z - 직각 좌표계 값을 사용하고 있다^[7].

기존 전투체계 소프트웨어 구조의 큰 변경 없이 제안한 알고리즘을 적용하기 위해서는 시선 속도를 (3)의 식에서 정의한 거리의 변화량($\frac{dr}{dt}$)이 아닌 매순간 센서로부터 수신되는 직각 좌표계 값인 V_x, V_y, V_z 을 이용하여 구할 수 있어야 한다. V_x, V_y, V_z 값을 이용하여 시선 속도를 구하기 위해 오일러 변환 공식을 사용한다^[8].

3차원 직각 좌표계에서 위치 정보는 다음과 같이 정의할 수 있다.

$$\mathbf{r} = x\mathbf{x} + y\mathbf{y} + z\mathbf{z}$$



[그림 8] 좌표 변환

$\mathbf{x}, \mathbf{y}, \mathbf{z}$ 은 직각 좌표축에 대한 단위 벡터이다.

V_x, V_y, V_z 값과 시선 속도와의 관계를 정의하기 위해 직각 좌표계에서 속도를 다음과 같이 정의할 수 있다.

$$\begin{aligned} \mathbf{v} &= \frac{d\mathbf{r}}{dt} = \frac{d[x\mathbf{x} + y\mathbf{y} + z\mathbf{z}]}{dt} \\ &= V_x \mathbf{x} + V_y \mathbf{y} + V_z \mathbf{z} \end{aligned}$$

그림 8과 같이 직각 좌표계를 구 좌표계로 변환하기 위한 오일러 변환 공식은 다음과 같다.

$$\begin{aligned} \begin{bmatrix} r \\ \theta \\ \phi \end{bmatrix} &= \begin{bmatrix} \cos(-\phi) & 0 & -\sin(-\phi) \\ 0 & 1 & 0 \\ \sin(-\phi) & 0 & \cos(-\phi) \end{bmatrix} \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \\ &= \begin{bmatrix} \cos \phi \cos \theta & \cos \phi \sin \theta & \sin \phi \\ -\sin \theta & \cos \theta & 0 \\ -\sin \phi \cos \theta & -\sin \phi \sin \theta & \cos \phi \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \end{aligned}$$

$\mathbf{x}, \mathbf{y}, \mathbf{z}$ 은 직각 좌표축에 대한 단위 벡터이며, \mathbf{r}, θ, ϕ 은 구 좌표축에 대한 단위 벡터이다.

구 좌표계에서 속도는 다음과 같이 정의할 수 있다.

$$\begin{aligned} \mathbf{v} &= \frac{d\mathbf{r}}{dt} = \frac{d[r\mathbf{r}]}{dt} = \left(\frac{dr}{dt}\right)\mathbf{r} + r\left(\frac{d\mathbf{r}}{dt}\right) \\ &= \left(\frac{dr}{dt}\right)\mathbf{r} + r \left[\left(\frac{\partial \mathbf{r}}{\partial \theta}\right)\left(\frac{d\theta}{dt}\right) + \left(\frac{\partial \mathbf{r}}{\partial \phi}\right)\left(\frac{d\phi}{dt}\right) \right] \\ &= \left(\frac{dr}{dt}\right)\mathbf{r} + r \cos \phi \left(\frac{d\theta}{dt}\right)\boldsymbol{\theta} + r \left(\frac{d\phi}{dt}\right)\boldsymbol{\phi} \end{aligned} \quad (4)$$

여기서, $\frac{\partial \mathbf{r}}{\partial \theta}$, $\frac{\partial \mathbf{r}}{\partial \phi}$ 는 오일러 변환 공식을 사용하여 다음과 같이 구할 수 있다.

$$\frac{\partial \mathbf{r}}{\partial \theta} = \frac{\partial [\cos \phi \cos \theta \mathbf{x} + \cos \phi \sin \theta \mathbf{y} + \sin \phi \mathbf{z}]}{\partial \theta} = -\cos \phi \sin \theta \mathbf{x} + \cos \phi \cos \theta \mathbf{y} = \cos \phi \boldsymbol{\theta}$$

$$\frac{\partial \mathbf{r}}{\partial \phi} = \frac{\partial [\cos \phi \cos \theta \mathbf{x} + \cos \phi \sin \theta \mathbf{y} + \sin \phi \mathbf{z}]}{\partial \phi} = -\sin \phi \cos \theta \mathbf{x} - \sin \phi \sin \theta \mathbf{y} + \cos \phi \mathbf{z} = \boldsymbol{\phi}$$

위의 식 (4)에서 첫 번째 항($\frac{dr}{dt}$)이 시선 속도값이 된다.

주어진 직각 좌표계의 속도값을 이용하여 시선 속도를 구하는 식은 다음과 같다.

$$\begin{aligned} \mathbf{v} &= V_x \mathbf{x} + V_y \mathbf{y} + V_z \mathbf{z} \\ &= \left(\frac{dr}{dt}\right)\mathbf{r} + r \cos \phi \left(\frac{d\theta}{dt}\right)\boldsymbol{\theta} + r \left(\frac{d\phi}{dt}\right)\boldsymbol{\phi} \\ \mathbf{v} &= \begin{bmatrix} V_x & V_y & V_z \end{bmatrix} \begin{bmatrix} \cos \phi \cos \theta & -\sin \theta & -\sin \phi \cos \theta \\ \cos \phi \sin \theta & \cos \theta & -\sin \phi \sin \theta \\ \sin \phi & 0 & \cos \phi \end{bmatrix} \begin{bmatrix} r \\ \theta \\ \phi \end{bmatrix} \end{aligned}$$

$$\therefore \frac{dr}{dt} = V_x \cos \phi \cos \theta + V_y \cos \phi \sin \theta + V_z \sin \phi \quad (5)$$

주어진 직각 좌표계의 속도값 V_x , V_y , V_z 를 이용하여 구한 시선 속도 식 (5)를 제안한 알고리즘 식 (1), (2)의 소프트웨어 구현에 사용하여 기존 소프트웨어 구조 변경을 최소화 하였다.

4. 시뮬레이션

3장에서 제안한 표적 융합 알고리즘의 성능을 시뮬레이션을 통해 확인하였다.

성능 평가에 사용된 센서 표적 융합의 기본 설정값은 다음과 같다.

- 표적 융합 최대 허용 위치 오차(α) : 1000m
- 표적 융합 최대 허용 속도 오차(γ) : 20m/s

시뮬레이션 단위는 거리의 경우 m, 속도의 경우 m/s이며, 시스템 표적, 센서 표적을 다음과 같이 정의하여 사용한다.

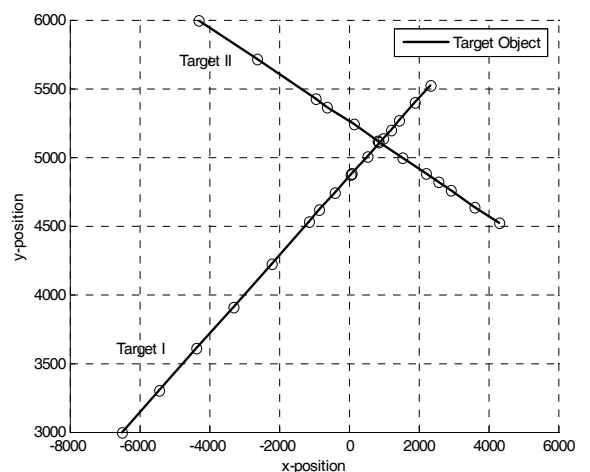
- 시스템 표적 : $S_1, S_2 \dots$ * 숫자 : 표적 번호
- 센서 표적 - 센서 A : $A_1, A_2 \dots$
- 센서 B : $B_1, B_2 \dots$

시뮬레이션에 사용된 이동하는 두 표적 정보는 다음과 같다.

[표 1] 표적 정보 초기값

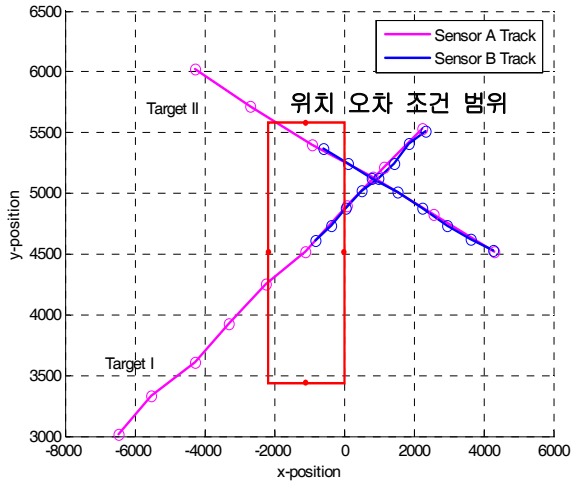
항목	표적 1	표적 2
position_x	-6500	-4300
position_y	3000	6000
position_z	1500	1700
velocity_x	210	320
velocity_y	60	-55
velocity_z	0	0
environment	Air	Air

표 1의 표적 초기값을 이용한 실제 표적의 궤적은 그림 9와 같다.



[그림 9] 실제 표적 궤적

그림 10은 각 센서 A, B에서 탐지한 센서 표적 정보를 나타낸다.



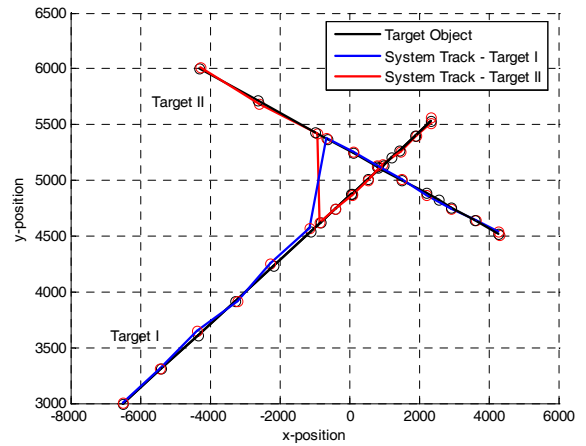
[그림 10] 탐지한 센서 표적 정보

시뮬레이션은 그림 10에서 보는 바와 같이, 새로운 센서 표적이 수신되어 시스템 표적과의 표적 동일성 검사가 이루어질 때, 표적 동일성 조건인 위치 오차, 식별 정보, 공간 범주 조건을 만족하는 시스템 표적이 여러 개가 존재하는 경우로 제한한다.

두 표적은 모두 센서 A에서 먼저 탐지하여 S_1 과 S_2 를 생성한다. 센서 B로부터 새로운 표적 B_1 과 B_2 가 시스템으로 수신되면, 시스템은 표적 동일성 검사를 통해 표적 융합을 수행하게 되는데, B_2 에 대해 S_1 과 S_2 모두가 표적 동일성 조건을 만족하게 되는 경우에 대한 시뮬레이션이다.

가. 기존 알고리즘

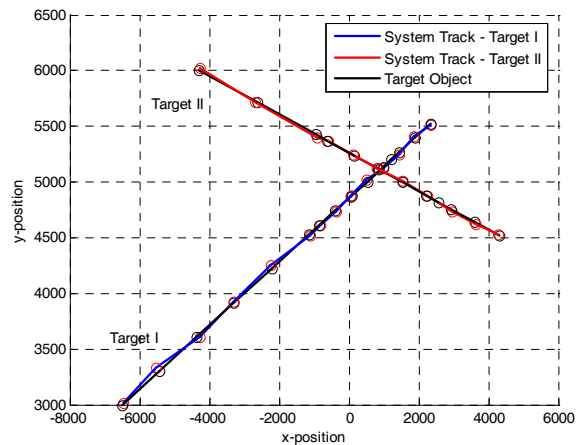
센서 표적에 대해 표적 동일성 검사를 만족하는 시스템 표적이 여러 개 존재할 경우, 시스템에서는 동일성 검사를 만족하는 시스템 표적이 존재하면 무조건 그 시스템 표적과 표적 융합을 수행하게 된다. 이러한 구조 때문에 B_2 에 대해 S_1 이 먼저 표적 동일성 조건을 만족하여 그림 11에서 보는 바와 같이 S_1 과 B_2 가 표적 융합되므로, 실제 표적의 위치와 전투체계에서 운용자에게 보이는 표적의 위치가 달라진다.



[그림 11] 시스템 표적 정보 - 기존 알고리즘

나. 제안 알고리즘

그림 12에서 보는 바와 같이, 제안 알고리즘에서는 B_2 에 대해 S_1 은 속도 오차 조건을 만족하지 못해 다른 표적으로 구분되고, S_2 는 제안 알고리즘의 모든 조건을 만족하여 B_2 와 표적 융합된다.



[그림 12] 시스템 표적 정보 - 제안 알고리즘

표적 동일성 검사를 수행할 때, 구체적으로 계산되어진 결과값들은 표 2와 같다. 첫 번째 표적 동일성 검사에서는 위치 오차 조건을 만족하지 못해 S_1 과 A_2 는 서로 다른 표적으로 인식되었다. 두 번째 표적 동일성 검사에서는 위치 오차 조건은 만족하지만, 속도 오차 조건 두 가지를 모두 만족하지 못해 S_1 과 B_2

[표 2] 표적 융합 시 계산되어진 결과값들

항목	세부 항목	1	2	3	4
시스템 표적	시스템 표적 번호	S ₁	S ₁	S ₂	S ₁
센서 표적	센서 표적 번호	A ₂	B ₂	B ₂	B ₁
위치오차	abs(dx)	1018.26	89.3928	23.6347	5.62904
	abs(dy)	2057.08	767.237	33.8283	9.29636
	기준값 [$\alpha + \beta * Abs(range)$]	1080.16	1072.94	1083.77	1072.98
속도오차	시스템 표적 시선 속도 v_{rs}	-26.9276	7.69102	-26.5255	8.11393
	센서 표적 시선 속도 v_{st}	-47.5244	-27.9637	-27.9637	6.35472
	abs(dVr)	20.5968	35.6547	1.43826	1.75921
	기준값 [$\gamma + \delta * Abs(v_{rs})$]	33.4638	23.8455	33.2627	24.057
융합 여부		X	X	O	O

는 서로 다른 표적으로 인식되었다. 세 번째와 네 번째 표적 동일성 검사는 제안 알고리즘의 모든 조건을 만족하여 각각 S₂와 B₂, S₁과 B₁이 표적 융합되었다.

5. 맺음말

함정 전투체계에 있어 주변 위협과 전술 상황을 정확하게 이해하고 대처하기 위해 방대한 양의 정보를 정확하게 처리하는 데이터 처리 능력은 필수적이다.

기존에는 하드웨어 및 센서의 제약으로 방대한 양의 정보를 처리하기 위해, 표적 융합에 있어 위치 정보, 공간 범주, 식별 정보, 센서 표적 정보만을 융합 판단 조건으로 이용했으나, 좀 더 정확한 표적 융합을 위해 본 논문에서는 시선 속도를 추가로 고려한 표적 융합 방법을 제안하였다.

또한 제안한 표적 융합 방법의 실제 적용을 위해 기존 전투체계의 소프트웨어 구조를 크게 변경하지 않는 범위 내에서 알고리즘을 구현할 수 있는 방법을 제시했으며, 이에 대한 성능을 확인했다.

향후에는 함정 전투체계에 적용하기 위해 다양한 표적 형태를 고려한 시스템 파라미터(최대 허용 위치 오차 상수나 최대 허용 속도 오차 상수 등)의 최적값 설정에 대한 연구가 추가로 수행되어야 할 것이다.

참 고 문 헌

- [1] Werner Henrich and Thomas Kausch, "Data Fusion for the Finnish Fast Attack Craft Squadron 2000 : Concept and Architecture", Processing of the seventh International Conference on Information Fusion, ISIF, pp. 842~849, 2004.
- [2] Werner Henrich and Thomas Kausch, "Data Fusion for the new German F124 Frigate Concept and Architecture", Processing of the sixth International Conference on Information Fusion, ISIF, pp. 1342~1349, 2003.
- [3] David L. Hall and James Llinas, "An Introduction to Multisensor Data Fusion", Proceedings of the IEEE, Vol. 85, No. 1, pp. 6~23, 1997. 1.
- [4] Qiao. Xiangdong and Wang. Baoshu, "A Modified Adaptive Track Fusion Approach", Processing of the fifth International Conference on Information Fusion, ISIF, pp. 1535~1541, 2002.
- [5] Ahmed Gad and M. Farooq, "Data Fusion Architecture for Maritime Surveillance",

Processing of the fifth International Conference on Information Fusion, ISIF, pp. 448~455, 2002.

- [6] 정영란, 신형조, “전투체계 지휘통제 설계기술 분석(표적관리분야)”, 국방과학연구소 기술보고서, 2004. 12.
- [7] TNNL, “Track Management Software Design Document”, 1998.
- [8] Richard L. PIO, “Euler Angle Transformations”, IEEE Transactions On Automatic Control, Vol. 11, Issue 4, pp. 707~715, 1966.