

論文

# 선박자동추적장치(ATA)의 목표물 추적 알고리즘 개발에 관한 연구

김석재\* · 구자윤\* · 윤수원\*

\*한국해양수산연수원 선박운항교육팀 교수

## A Study on the Development of Tracking Algorithm for Shipborne Automatic Tracking Aids

Seok-Jae Kim\*, Ja-Yun Koo\*, Su-Weon Yoon\*

\*Nautical Science Team, Korea Institute of Maritime and Fisheries Technology, Busan, 608-080, Korea

**요 약 :** 국제해사기구(IMO)는 해상인명안전협약(SOLAS)에 2002년 7월 1일부터 신조되는 총톤수 500톤 이상의 모든 선박에 자동 선박추적장치(ATA)를 탑재시키도록 규정하였으나, 이 SOLAS 협약에 적용되지 않는 현재 운항중인 10,000톤 미만의 현존선은 ATA의 비탑재로 인하여 충돌사고가 빈발하고 있다.

본 논문에서는 ATA의 일부 요소기술이 국산화되어 있으나 가장 핵심적인 자동추적 알고리즘 개발은 아직 미비한 실정에 있으므로 자동추적 알고리즘의 핵심요소기술의 연구개발을 통하여 ATA를 국산화하여 연안에서 항해하는 중소형 선박에 보급함으로써 충돌사고 등을 미연에 방지하고자 한다.

**핵심용어 :** 물표, RSC, 자동추적장치(ATA), 레이더, 트랙, 상관, 게이트, 윈도우, 최근접 데이터 선택

**ABSTRACT :** *Ships of 500 gross tonnage and upwards constructed on or after 1 July 2002 shall have an automatic tracking aids according to SOLAS V/19 but existing ships less than 10,000 gross tonnage constructed before 1 July 2002 have potential collision risks due to the lack of automatic plotting devices like as an ATA.*

*This paper aims to provide a homemade ATA by developing the tracking algorithm for ATA and to prevent collision incidents by distributing ATA system to coasters.*

**KEY WORDS :** *Target, Radar Scan Converter, Automatic Tracking Aids, Radar, Track, Correlation, Gate, Window, Nearest Neighbor Rule*

### 1. 서 론

선박자동추적장치(ATA)는 운항중인 선박에서 레이더 신호를 RSC로 변환, 입력받은 신호를 처리하여 주위에 운항하고 있는 선박들과 각종 물표들을 자동으로 축출 인식하고 상대선의 움직임을 자동 추적할 수 있는 기능을 가짐으로서 다른 선박 및 각종 물표와의 충돌이나 좌초 또는 악천후에 의한 조난 등을 미연에 예방할 수 있도록 정보를 제공하는 장치이다.

국제해사기구(IMO)는 해상인명안전협약(SOLAS)에 2002년 7월 1일부터 신조되는 총톤수 500톤 이상의 모든 선박에 ATA를 탑재시키도록 규정하였으나, 이 SOLAS 협약에 적용되지 않는 현재 운항중인 10,000톤 미만의 현존선은 ATA의 비탑재로 인하여 충돌사고가 빈발하고 있다.

선박용 레이더의 표시장치에서 나타나는 다양한 물표의 움직임을 예측하기 위하여 10,000톤 이상의 대형선박은 자동 플

로팅 기능을 가지는 자동 레이더 플로팅 장치(ARPA)를 이용

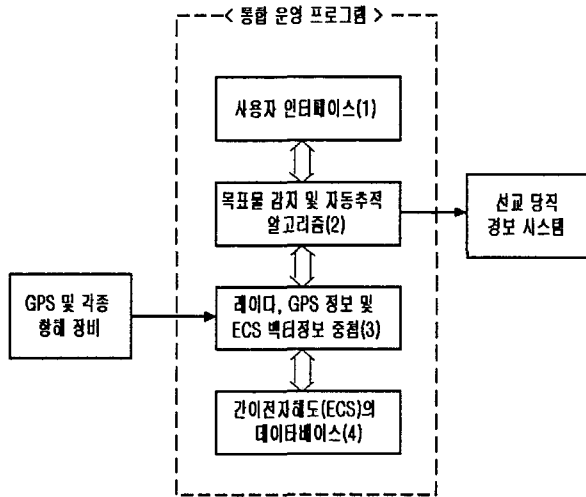
하여 물표의 정보를 습득하지만, 10,000톤 미만의 중소형 선박은 레이더에 ARPA 기능을 가지고 있지 않고, 소형 레이더 스크린 때문에 수동 플로팅을 위한 플로터도 확보하지 못하고 있기 때문에 타선의 움직임을 정확히 파악하는데 한계를 가지고 있다. 또한 항해사의 상대적 피로와 능력 저하로 선박충돌 사고나 좌초의 위험에 항상 노출되어 있기 때문에 레이더 화면에 나타나는 여러 물표를 자동 축출하여 인식하고 그 인식된 물표의 움직임을 예측할 수 있는 ATA는 필수적인 항해장비라 할 수 있다.

본 논문에서는 ATA에서 가장 핵심적인 기술인 목표물 자동추적 알고리즘을 연구 개발하여 실제 장비에 적용하여 실효성을 확인하고자 한다. 따라서 제안된 자동추적 알고리즘이 적용된 ATA를 연안에서 항해하는 중소형 선박에 보급함으로써 충돌사고를 미연에 방지하여 해상안전을 확보하기 위한 것이다.

제안된 ATA의 운영프로그램의 포괄적인 구성도를 그림 1에서 나타내었다. 여기에서 사용자 인터페이스는 ATA를 사용하는 운영자가 요구하는 각종 입력 명령을 해독하여 일차적으로 처리운영하고, 목표물 축출 감지 및 자동 추적 알고리즘은 레이더 표시화면에서 나타나는 각종 물표 데이터에서 목표물

을 자동으로 추출하고 추출한 목표물의 앞으로 위치, 속도 등을 지속적으로 예측하여 추적하는 알고리즘으로 ATA의 핵심적인 요소 기술로 본 논문에서 연구 제안할 것이다. 그리고, 레이다, GPS 정보 및 간이전자해도 벡터정보의 중첩은 레이다의 화면표시정보와 간이전자해도(ECS)의 데이터베이스에서 레이다 화면표시 영역의 데이터를 GPS의 위치정보 및 각종 항해장비에서 입력된 데이터를 참조, 보정하여 중첩시켜 화면에 나타내는 프로그램이고, 마지막 간이전자해도(ECS)의 데이터베이스(4)는 국가에서 인정한 전자해도(ENC)를 소형 선박에서 쉽고 원활하게 사용할 수 있도록 축소시킨 간이전자해도를 구성하는 프로그램으로 구성된다.

따라서, 본 논문은 목표물의 자동추적의 개념을 먼저 설명하고, 추적된 목표물의 데이터의 구조를 설계하고 추적알고리즘 및 위치추정에 대한 정확도를 높이기 위한 알고리즘을 제안하여 결론을 맺는 것으로 서술하였다.

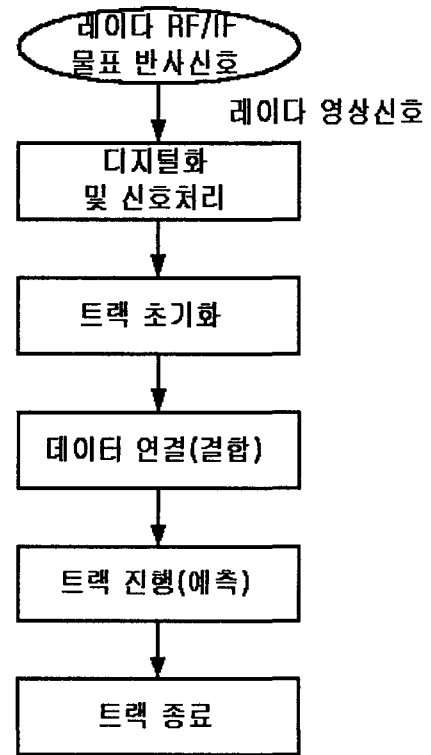


<그림 1> ATA의 운영프로그램의 구성도

## 2. 추적목표물의 데이터 구조설계 및 적용

레이다에서 물표의 플로팅을 자동으로 행하기 위해서는 레이다로 검출한 물표 중에서 타선 등 이동물표와 같은 위험 목표물을 확실하게 추출하고, 그 후 시간의 경과에 따라 얻어지는 새로운 레이다 신호 중에서 앞에서 포착한 동일 목표물을 식별하고 이러한 식별을 계속 추적, 유지하여야 한다. 즉 목표물의 이동을 예측하고, 입력되는 신호의 위치를 구하는 추적 동작이 있어야 한다. 따라서 ATA에서 가장 중요한 알고리즘은 식별된 목표물의 추적(Tracking) 알고리즘의 구현이고, 또한 이 알고리즘에서 입력되는 데이터에서 추적동작을 계속적으로 유지하기 위해서는 선별된 위험 목표물에 대한 데이터 구조에 대한 설계와 연결된 데이터 사이에 결합(Link)방법이 중요하게 부각된다. 여기에 위험 목표물의 추적 알고리즘의 기본적

인 동작을 간단히 설명할 수 있는 블록도를 그림 2에 나타내었다. 레이다에서 입력된 반사신호, 즉 레이다의 영상신호는 RSC를 통하여 디지털로 변환되고 기본적인 잡음제거 및 물표를 검출하는 신호처리 동작으로 목표물의 위치정보를 구하게 된다. 이 위치정보를 이용하여 추적동작을 하기 위해 트랙을 생성하는 초기화 동작을 하게 되고, 이 트랙은 이전에 생성된 트랙과 연관성을 검토한 후 데이터 결합을 한 뒤 이 목표물이 앞으로 어느 곳으로 움직일 것인가를 예측하는 트랙진행 동작을 하고 나면, 주어진 위치 정보 데이터에 대한 추적 알고리즘의 동작이 완료되는 트랙 종료 동작이 이루어 지는데, RSC에 의해 주어진 모든 물표의 위치정보에 대해 그림 2와 같은 동작이 반복하게 된다.



<그림 2> ATA의 추적 알고리즘 동작 블록도

### 1) Track-While-Scan(TWS) 시스템

선박자동추적 동작을 하는 데이터처리는 레이다 안테나의 한 스캔동안 추적동작이 이루어지는 TWS(Track-While-Scan) 운용 알고리즘으로 레이다로부터 입력, 처리되어 주어진 물표의 위치 정보를 이용하여, 기존의 물표는 트랙을 갱신하고, 새로운 물표에 대해서는 새로운 트랙을 생성하는 동작을 한다. 여기에서 트랙이라는 것은 동일한 물표에 대한 정보(물표 식별번호, 위치, 속도 등)의 집합을 의미한다. 이 트랙의 데이터를 처리하기 위해 다음과 같은 4가지 데이터 저장 구조를 사용한다.

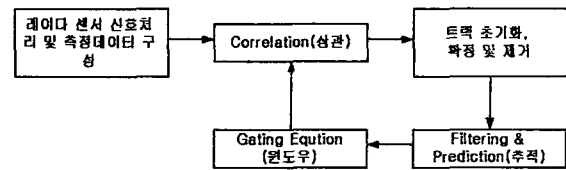
- 입력 트랙(input track)
- 임시 트랙(tentative track)
- 확정 트랙(firm track)
- 제거 트랙(deleted track)

TWS 실행동작 알고리즘을 세분화하여 순차적으로 설명하면 다음과 같다.

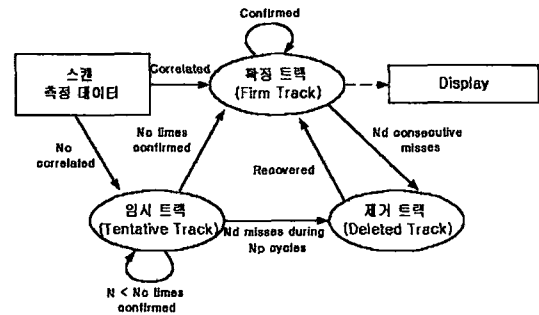
첫번째 과정에서는 레이더로부터 입력된 데이터가 기존의 트랙과 연관되는 여부에 따라 입력된 데이터를 임시 트랙에 보내거나 확정 트랙에 보낸다. 이때 입력된 데이터가 기존의 트랙과 연관성이 있는지의 판단은 각 트랙이 규정하는 일정한 영역(윈도우, window)내에 입력 데이터가 존재하는지에 따라 결정되며 각 윈도우의 중심은 물표의 위치에 대한 예측값이 된다.

두번째 과정에서는 임시 트랙에 저장되어 있는 데이터를 사용하여 기존의 트랙을 갱신하는 과정이다. 한 개의 트랙은 매 스캔(scan)마다 갱신할 수 있으며, 트랙의 갱신은 그 트랙에 연관되는 모든 데이터가 레이더에 의해 제공될 때까지 이루어지지 않는다. 임시의 트랙에 대한 갱신은 안테나의 방위각이 그 트랙에 대한 방위각을 어느 정도 지나쳤을 때 시작된다. 만약 갱신과정이 시작되었을 때 해당 트랙에 연관되는 데이터가 없다면 해당 물표는 일정한 속도로 움직인다고 가정하여 트랙을 갱신한다. 만약 갱신과정이 시작되었을 때, 해당 트랙과 연관되는 데이터가 여러 개가 있다면, 해당 트랙과 가장 연관성이 많은 데이터를 선택하여 트랙을 갱신하며, 이 때 데이터의 선택 과정은 임시 트랙에 저장되어 있는 데이터 중에서 해당 트랙에 가장 가까운 데이터를 선택한 다음, 그 데이터가 미리 설정된 윈도우 내부에 있는 지 판별하여, 그 내부에 있으면 트랙의 갱신에 사용되는 데이터로 선택하는 것이다. 여기에서 사용되는 윈도우의 중심은 예측된 물표의 위치이고, 윈도우의 크기는  $\alpha$ - $\beta$  추적 필터에서 물표의 다음 위치를 예측하는데 사용되는 예측 오차의 공분산값에 따라 결정된다. 이런 과정을 거쳐서 1개의 데이터가 선택되면, 해당 물표의 가속도 변화가 있었는지 여부를 판별하기 위해 미리 설정한 가속도 유무 판별용 윈도우를 사용한다. 선택한 데이터가 가속도 유무 판별용 윈도우의 내부에 있는지, 외부에 있는지에 따라 이에 상응하여  $\alpha$ - $\beta$  추적 필터의 이득  $\alpha$ ,  $\beta$ 의 값이 다른 필터를 사용함으로써 해당 물표의 가속도에 따른 예측 및 추정 오차를 보상한다.

세번째 과정은 새로운 물표에 대한 트랙을 생성하는 과정이다. 생성 후에 저장된 데이터는 한 스캔이 경과하면 제거 트랙으로 이동하게 되고, 제거 트랙에 저장된 데이터는 저장된 기간이 스캔 시간의 3.5배 ~ 4배를 초과하면 제거된다. 확정 트랙에 저장된 데이터는 제거 트랙에 저장된 데이터와 연관되어 하나의 트랙을 생성시킬 수 있다.



<그림 3> 데이터 처리기 알고리즘의 순환도



<그림 4> 데이터 처리기 알고리즘의 논리도

네번째 과정은 기존의 물표에 대한 트랙을 제거하는 과정이다. 이러한 과정은 기존의 물표가 레이더의 탐지 영역을 벗어나거나 물표가 가려지는 경우에 발생할 수 있다. 이 TWS 시스템에서 트랙의 소멸 요건은 연속적인 4번의 스캔에서 해당 트랙과 관련된 데이터가 발견되지 않을 때 트랙을 제거하는 것으로 한다.

위에서 기술한 내용을 그림 3과 그림 4와 같이 간단히 나타낼 수 있다. 레이더 시스템으로부터 입력된 반사파 데이터를 처리하여 측정된 목표물 데이터를 자동으로 추적하기 위해서는 그림 4와 같은 과정을 순환하여 처리하게 된다.

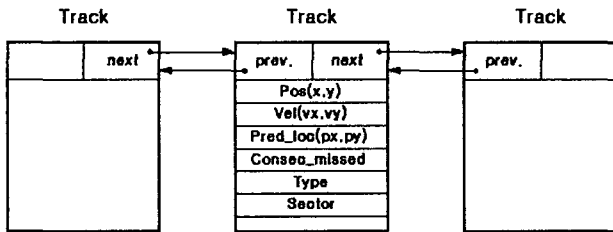
## 2) 트랙 데이터 저장구조 설계와 그룹별 트랙의 관리

일반적으로 ATA에서 추적된 물표에 대해 주어지는 정보는 물표의 현재 거리, 물표의 현재 방위, CPA에서 예측된 물표 거리, TCPA, 물표의 계산된 선수 방향, 물표의 계산된 속도 등이다. 레이더 신호를 RSC 및 신호처리부로부터 받은 데이터(물표의 위치, 시간 등)를 이용하여 트랙(track)을 형성시키고, 시스템 제어부와 서로 신호를 주고 받으면서 각 경우에 따라 필요한 데이터를 처리한다. 같은 트랙 내에 존재하는 데이터들의 연결구조와 데이터 저장구조를 다음 그림 5와 같이 이중연결구조로 설계하여 연관된 데이터의 순서와 연결성을 보장하는 데이터베이스 구조로 설계한다. 또한, 그림 6과 같이 저장된 트랙 데이터베이스에서 트랙의 효율적인 데이터 관리를 위해 그림 6과 같은 몇 개의 그룹별로 트랙을 관리하는 방안 적용한다. 그림 6에서는 트랙을 확정(Firm), 안정(Stationary), 제거(Deleted) 트랙에 대해서만 예시적으로 보여주고 있으며 각 그룹별로 등록된 트랙을 관리하는 데이터베이스의 테이블 및 각 트랙을 연결을 관리하는 테이블 사이의 연관성을 나타내고 있다. 그리고, 추적되는 위험물표의 데이터

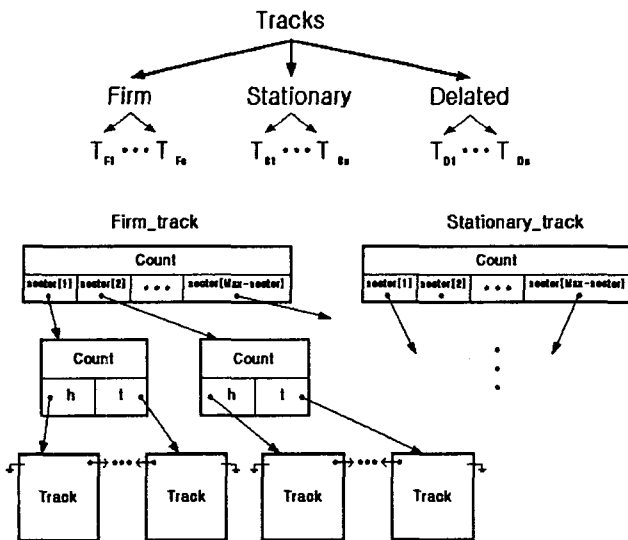
트랙에 대한 데이터 및 파일 관리는 초기화, 결합, 업데이트, 분기, 제거 등의 5개의 상태로 구분하여 처리하고, 또한 데이터베이스 관리를 위한 초기화, 삽입, 제거, 검색/비교, 정렬 등의 명령을 수행할 수 있도록 한다.

Table Name : Track/Plot

- item : ID.
- Position (x,y)
- Velocity (vx,vy)
- Predicted Position (px, py)
- No. of consecutive misses
- No. of consecutive success
- Track type (Tentative, Firm, Deleted, ..)
- Sector
- Pointer (previous, next)
- etc. ...



<그림 5> 추적 물표 데이터베이스의 저장 구조



<그림 6> 그룹화된 트랙의 데이터베이스 테이블의 저장상태

### 3. 목표물 추적 알고리즘의 기능 구현

#### 1) 상관 알고리즘(Correlation Algorithm)

게이터의 윈도우(window)는 트랙의 예측된 위치를 중심으로 일정한 영역을 나타내는 것으로서, 레이더로부터 입력되어 처리된 물표의 위치가 기존에 생성되어 있는 트랙의 윈도우 중에서 어느 하나에 속하게 되면 그 물표는 그 트랙과 상관관계가 있다고 판단하고, 만약 어느 트랙의 윈도우에도 속하지 않으면 그 물표는 새로운 트랙을 만드는 데 사용된다. 만약 어느 하나의 트랙과 여러 개의 물표가 상관관계가 있게 되거나, 한 개의 물표가 여러 트랙과 상관관계가 있는 경우에는 최종적인 상관관계를 정하기 위해 상관규칙이 필요하다. 본문에서 제안하여 사용한 상관규칙은 최근접 데이터 선택규칙(Nearest Neighbor Rule)과 모든 근접 데이터 선택규칙(All Neighbor Rule)이다. 그리고, 일반적으로 사용되는 윈도우는 사각 윈도우(Rectangular Window)와 타원 윈도우(Ellipsoidal Window)가 있는데, 여기에서는 연산속도가 빠른 사각 윈도우를 사용하였다.

물표의 위치정보는 거리(Range)와 방위각(Azimuth)으로 나타내는데, 물표의 위치정보와 기존의 트랙간의 상관관계를 결정하는데 사용되는 윈도우의 크기는 다음과 같이 정의한다. 거리에 대한 윈도우의 최대값과 최소값을 각각 RWMAX, RWMIN이라고 하면, 다음 식(1), (2)가 된다.

$$RWMIN = D - 3\sqrt{P_r(1,1)} - [\gamma^2 \frac{T^4}{3}]^{1/2} \dots (1)$$

$$RWMAX = D + 3\sqrt{P_r(1,1)} + [\gamma^2 \frac{T^4}{3}]^{1/2} \dots (2)$$

여기에서  $D$ 는  $P_r(1,1)$ 은 거리의 예측 오차에 대한 분산이다. 그리고,  $T$ 는 표본화 시간이고,  $\gamma$ 는 가속도 변화에 대한 표준편차이다.

마찬가지로 방위각에 대한 윈도우의 최소값과 최대값을 BWMIN, BWMAX라고 하면, 다음 식(3), (4)가 된다.

$$BWMIN = \hat{B} - 3\sqrt{P_{az}(1,1)} - \frac{[\gamma^2 \frac{T^4}{3}]^{1/2}}{\hat{D}} \dots (3)$$

$$BWMAX = \hat{B} + 3\sqrt{P_{az}(1,1)} + \frac{[\gamma^2 \frac{T^4}{3}]^{1/2}}{\hat{D}} \dots (4)$$

여기에서  $B$ 는 방위각의 예측값이고,  $P_{az}(1,1)$ 는 방위각의 예측오차에 대한 분산이다.

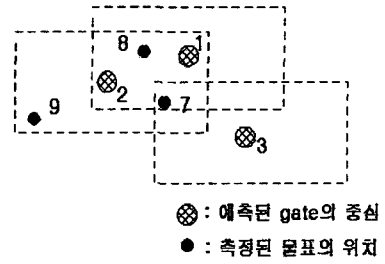
따라서 상관 알고리즘은 측정치와 트랙을 연관시켜주는 방법으로 게이터에 의한 출력 데이터와 기존의 트랙 사이의 결합(Assignment)을 시켜주는 과정이다. 여기에서 게이터란 윈도우(분산치 내) 내에 속하는지 아닌지 즉, 레이더의 측정 데이터가 기존에 추적되고 있는 물표에 속하는지 아닌지를 결정하는 동작을 의미한다. 본문에서 사용한 상관 알고리즘은 다음과 같은 두 가지 방법을 사용하였으며 3가지 상황별로 상관관계의 판단 알고리즘의 기준을 제시한다.

① 최근접 데이터 선택 방식(Nearest Neighbor Approach) - Simple

다수의 레이다 데이터가 하나 혹은 여러 개의 윈도우에 중복해서 속해 있는 경우 유일한 연관 방법을 실현시켜주는 방법으로 NN 방식이라 불리우는 통계적 거리가 가장 작은 데이터를 선택하는 가장 간단한 방식이다.

② 모든 근접 데이터 선택 방식(All Neighbor Approach) - Probabilistic

윈도우 내에 들어 있는 다수의 데이터를 모두 추적에 사용하는 방법으로 조금 연산량이 많아지고 복잡한 단점이 있으나 추적의 정밀도를 높여주는 방식으로 AN 방식 혹은 PDA 필터라 불리우는 확률을 사용하는 방식이다.



measurement	1	2	3
7	4.2	5.4	6.3
8	1.2	3.1	X
9	X	7.2	X

<그림 7> 상황 3의 예시상황과 측정 데이터-트랙의 거리계산 행렬(표)

- 상황 1. 한 게이트 내에 하나의 측정 데이터가 존재하는 경우 : ⇨ 평범하게 해당 트랙으로 바로 연관
- 상황 2. 한 gate 내에 여러 개의 측정 데이터가 존재하는 경우 : ⇨ 예측위치에서 가장 가까운 측정 데이터를 그 트랙의 추적 유지에 사용
- 상황 3. 여러 gate에 여러 개의 측정 데이터가 서로 중복되게 속해 있는 경우 : ⇨ 그림 7과 같은 경우에 측정 데이터와 트랙 사이의 거리를 계산한 행렬(표)에 의하여 처리하는 과정과 같은 알고리즘으로 구성

▷ 알고리즘 1 :

- ① 중복된 게이트에 포함된 측정 데이터에 대한 예측된 위치 중심에서의 거리를 계산한 행렬(표)를 작성한다.
- ② 각 게이트에 대해 가장 가까운 측정 데이터를 선택한 후 선택된 측정 데이터와 연관시킨 트랙은 제거한다.
- ③ 나머지 트랙의 게이트에 대해 순서 ②를 반복한다.

▷ 알고리즘 2 :

- ① 중복된 게이트에 포함된 측정 데이터에 대한 예측된 위치 중심에서의 거리를 계산한 행렬(표)를 작성한다.
- ② 게이트 중 단 하나의 측정 데이터만이 포함된 측정 데이터와 트랙의 연결은 모두 제거한다.
- ③ 각 게이트에 대해 가장 가까운 측정 데이터를 선택한 후 선택된 측정 데이터와 연관시킨 트랙은 제거한다.
- ④ 나머지 트랙의 게이트에 대해 순서 ③을 반복한다.

2) 예측(Prediction) 및 여파(Filtering)

본 논문의 예측 및 여파에서 사용한 기초적인 알고리즘은  $\alpha$ - $\beta$  추적 필터이다. 이 추적필터는 레이다 데이터 처리기에서 중요한 기본적 요소로서 ①물표의 속도와 가속도 추정 ②물표의 미래위치 예측 등의 기능을 얻기 위해 물표에 대한 데이터의 측정치(거리, 방위)를 이용하여 처리한다.

(1) 물표추적(상태추정)

물표추적 혹은 상태추정(State Estimation)은 기존 데이터 및 새로운 물표 데이터로부터 데이터의 각 스캔 후에 이루어진다. 물표추적기는 동적운동모델로 가정하고 데이터의 측정치에 근거하여 위치, 속도, 좌표 파라메타를 추정하게 된다. 이러한 추정치는 다음 스캔에서 레이다에 의해 측정될 위치에 있는 물표의 위치를 예측하는데 사용한다. 따라서 상태추정은 시스템 상태와 연관되는 측정 데이터  $z$ 가 주어진 상태에서 물표상태  $x$ 를 결정하는 문제이다. 일반적으로 측정모델은 다음 식(5)와 같은 관계로 연관되며, 여기서  $v$ 는 측정잡음으로서 보통 가우시안으로 가정한다.

$$z = f(x) + v \dots \dots \dots (5)$$

일반적으로 순환적 추정기(Recursive Estimation)는 다음과 같은 세가지 기능을 수행하는데 사용될 수 있다.

- ① 여파(Filtering) : 시간  $t$ 이전의 측정 데이터  $Z$ 가 주어진 경우 시간  $t$ 에서의 물표상태  $X$ 의 추정.
- ② 예측(Prediction) : 시간  $t$ 이전의 측정 데이터  $Z$ 가 주어진 경우 시간  $t+k$ 에서의 물표상태  $X$ 의 추정.
- ③ Smoother : 시간  $t$ 전후에 측정 데이터  $Z$ 가 주어진 경우 시간  $t$ 에서의 물표상태  $X$ 의 추정.

따라서, 순환적 물표추적은 다음과 같은 기능을 요구하는 하나의 상태추정 문제로 귀결된다. 즉, 시간  $t$ 까지의 측정데이터를 통해 시간  $t$ 에서의 물표상태 벡터를 추정하기

위해 필요한 데이터처리 필터와 물표상태를 미래의 샘플 시간으로 추정하는 예측기로 이루어진다.

(2) 추적필터

선형운동을 하는 물표에도 추적을 위해 앞에서 언급한 최근접 데이터 선택규칙 알고리즘을 이용한다. 이는 현재의 측정데이터를 가지고 예측한 위치를 중심으로 허용 원도우내에서 확률적인 거리가 가장 가까운 부근에서 검출된 측정데이터를 가지고 물표의 위치나 속도 등으로 구성된 상태벡터를 갱신(update)하여 물표에도 추적을 하는 방법이다. 일반적으로 확률적인 거리는 다음 식(6)로 주어진다.

$$D^2 = \frac{(r_p^2 - r_m^2)}{\sigma_r^2} + \frac{(\theta_p^2 - \theta_m^2)}{\sigma_\theta^2} \dots \dots \dots (6)$$

여기에서  $(r_p, \theta_p)$ 는 예측 위치이고,  $(r_m, \theta_m)$ 은 측정 위치를 나타낸다. 그리고,  $\sigma_\theta^2$ 은  $(\theta_p - \theta_m)$ 의 분산이고,  $\sigma_r^2$ 은  $(r_p - r_m)$ 의 분산이다.

(3) α-β 추적 필터 알고리즘

항해용 레이더를 이용하여 물표를 추적하는 경우 대부분의 물표가 고정된 지형지물이거나 느린 속도로 진행하는 선박들이기 때문에 이러한 물표에 대해서 적용되는 추적 알고리즘은 시불변(Time-invariant)시스템을 가정하여 상태추정을 하게된다. 따라서, 여기에서 사용하고 있는 α-β 추적 필터는 다음과 같은 운동모델에 근거하고 있다.

$$\mathbf{x}(k+1) = \mathbf{F}\mathbf{x}(k) + \mathbf{v}(k) \dots \dots \dots (7)$$

$$\mathbf{z}(k) = \mathbf{H}\mathbf{x}(k) + \mathbf{w}(k) \dots \dots \dots (8)$$

위 식(7), (8)에서  $\mathbf{x}(k)$ ,  $\mathbf{F}$ ,  $\mathbf{H}$ 는 다음과 같다.

$$\mathbf{x}(k) = \begin{bmatrix} x_1(k) & x_2(k) \end{bmatrix}^T \dots \dots \dots (9)$$

$$\mathbf{F} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \quad \mathbf{H} = [1 \quad 0] \dots \dots \dots (10)$$

여기에서 T는 표본화 주기(sampling interval)을 나타낸다. 그리고  $\mathbf{v}(k)$ 와  $\mathbf{w}(k)$ 는 각각 프로세스 잡음과 측정잡음을 나타내며 평균값이 0이고 가우시안(gaussian) 분포를 가진 것으로 모델링하여 사용한다. 이들에 대한 공분산(covariance)은 다음과 같다.

프로세스잡음 공분산 :  $E[\mathbf{v}(k)\mathbf{v}'(j)] = \mathbf{Q}\delta_{kj}$

측정잡음 공분산 :  $E[\mathbf{w}(k)\mathbf{w}'(j)] = \mathbf{R}\delta_{kj} = \sigma_w^2 \delta_{kj}$

위에서 정의한 운동모델은 거리의 변화나 방위각의 변화를 모델링하는데 사용되며, 식(9)에서 살펴보면 거리의 변화를 모델링하는 경우에  $x_1(k)$ 는 시간 k에서의 거리가 되

고,  $x_2(k)$ 는 시간 k에서의 거리 변화율을 의미한다. 방위각의 변화를 모델링하는 경우에도 같은 방법으로 적용된다.

α-β 추적 필터의 실행은 크게 3가지로 구분하여 예측과정, α-β 추적 필터의 이득 계산과정, 필터링 과정으로 나눌 수 있다.

① 예측 과정

$$\hat{\mathbf{x}}(k+1|k) = \mathbf{F}\hat{\mathbf{x}}(k) \dots \dots \dots (11)$$

$$\mathbf{P}(k+1|k) = \mathbf{F}\mathbf{P}(k|k)\mathbf{F}' + \mathbf{Q} \dots \dots \dots (12)$$

위 식(11)은 k번째 스캔(scan)까지의 측정데이터가 주어졌을 때 (k+1)번째 스캔에서의 상태를 예측한 값이고, 식(12)는 k번째 스캔까지의 측정데이터가 주어졌을 때 (k+1)번째 스캔에서의 예측 오차 공분산(prediction error covariance)을 의미한다.

$$\hat{\mathbf{z}}(k+1|k) = \mathbf{H}\hat{\mathbf{x}}(k+1|k) \dots \dots \dots (13)$$

$$\mathbf{S}(k+1) = \mathbf{H}\mathbf{P}(k+1|k)\mathbf{H}' + \mathbf{R} \dots \dots \dots (14)$$

위 식(13)은 (K+1)번째 스캔에서의 예측된 상태값으로 (k+1)번째의 측정값을 예측하는 것을 의미하고, 식(14)는 (k+1)번째의 측정값이 입력되었을 때 예측값과 실제 측정값의 차이(즉 innovation)에 대한 공분산(covariance)이고,  $\mathbf{R}$ 은 (k+1)번째 스캔에서의 측정오차 공분산(measurement error covariance)을 의미한다.

② α-β 추적 필터의 이득 계산과정

시스템의 운동모델에서 계수는 시간과 상관없이 일정하기 때문에 필터 이득은 시불변하게 결정된다. 이는 상태 추정 공분산(state estimation covariance)이 안정된 상태값(steady-state value)으로 수렴함을 의미한다. 따라서 이를 식으로 표현하면 다음 식(15)가 되고, 상태 예측 공분산은 식(16)으로 표시된다.

$$\lim_{k \rightarrow \infty} \mathbf{P}(k|k) = \lim_{k \rightarrow \infty} \mathbf{P}(k+1|k+1) = [p_{ij}] \dots (15)$$

$$\lim_{k \rightarrow \infty} \mathbf{P}(k+1|k) = [m_{ij}] \dots \dots \dots (16)$$

필터 이득은 다음 식(17)과 같다.

$$\lim_{k \rightarrow \infty} \mathbf{W}(k) = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta/T \end{bmatrix} \dots \dots \dots (17)$$

따라서 식(14)는 다음 식(18)이 되고, 필터 이득  $\mathbf{W}$ 는 다음 식(19)로 주어진다.

$$\mathbf{S} = \mathbf{H} \begin{bmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{bmatrix} \mathbf{H}' + \mathbf{R} = m_{11} + \sigma_w^2 \dots \dots (18)$$

$$\mathbf{W} = \mathbf{P}\mathbf{H}'\mathbf{S}^{-1}$$

$$= \begin{bmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{bmatrix} \mathbf{H}\mathbf{S}^{-1} = \begin{bmatrix} \frac{m_{11}}{m_{11} + \sigma_w^2} & \frac{m_{12}}{m_{11} + \sigma_w^2} \end{bmatrix} (19)$$

식(17)과 식(19)를 비교하면,  $g_1$ 과  $g_2$ 는 각각 식(20), (21)이

된다.

$$g_1 (= \alpha) = \frac{m_{11}}{m_{11} + \sigma_w^2} \dots \dots \dots (20)$$

$$g_2 (= \beta/T) = \frac{m_{12}}{m_{11} + \sigma_w^2} = g_1 \frac{m_{12}}{m_{11}} \dots \dots \dots (21)$$

③ 필터링 과정

(k+1)번째에서 갱신된 상태 공분산(updated state covariance)은 식(22)로 표시되고, 이는 식(23)으로 된다.

$$P(k+1|k+1) = [I - \mathbf{WH}]P(k+1|k) \dots \dots (22)$$

$$\begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} = (I - \mathbf{WH}) \begin{bmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{bmatrix}$$

$$= \begin{bmatrix} (1 - g_1)m_{11} & (1 - g_1)m_{12} \\ (1 - g_1)m_{12} & m_{22} - g_2 m_{12} \end{bmatrix} \quad (23)$$

(k+1)번째에서 갱신된 상태 추정(updated state estimation)은 식(24)가 되고, 이것이 2차원 모델에서의  $\alpha$ - $\beta$  추적 필터로 알려져 있다.

$$\mathbf{x}(k+1|k+1) = \mathbf{x}(k|k) + \begin{bmatrix} \alpha \\ \beta/T \end{bmatrix} [z(k+1) - z(k+1|k)] \quad (24)$$

식(12)를 다시 표현하면, 식(25)가 된다.

$$P(k|k) = \mathbf{F}^{-1}[\mathbf{P}(k+1|k) - \mathbf{Q}](\mathbf{F}^{-1})' \dots \dots (25)$$

3)  $\alpha$ - $\beta$  추적 필터 적용과 물표 위치인식 알고리즘

(1)  $\alpha$ - $\beta$  추적 필터 적용

선박용 레이다에서 목표물을 추적하는데 널리 이용되는 추적 알고리즘은  $\alpha$ - $\beta$  추적 필터와 칼만 필터가 널리 이용되고 있다. 특히  $\alpha$ - $\beta$  추적 필터방법은 선박처럼 움직이는 속도에 비해 주어지는 데이터 양이 많은, 즉 데이터 비(Data Rate)가 높은 경우나 가속도에 의한 움직임이 거의 없기 경우에 정도가 높은 추적 결과를 나타내기 때문에 본 논문에서도 이용하였다. 또한  $\alpha$ - $\beta$  추적 필터는 개념이 간단하고 소모 메모리가 적으며 처리시간이 짧다는 이점도 있다.

레이다로부터 얻어진 위치 데이터에는 레이다 자체의 성능에 기인하는 것과 반사파의 변동에 기인하는 오차가 포함되어 있으며, 또한 물표의 앞으로 위치 예측이나 물표의 실제속도 산출에 사용되는 본선의 침로 및 속력 데이터에도 이러한 오차가 포함되므로 목표물의 예측위치는 당연히 어긋나게 된다. 따라서 추적 필터처리는 다음과 같은 방법으로 행하여진다.

- ① 예측한 위치와 실제 관측한 위치와의 차이를 구하고, 이 분산을 평균해서(평활화) 현재의 물표위치(평활위치)를 설정한다.
- ② ①과 같은 방식으로 앞전의 평활위치로부터 이번의 평활위치까지의 속도(평활속도)를 산출하고, 이를 추적 물표의 속도로 가정해서 물표의 현재 평활위치로부터

앞으로의 위치를 예측한다.

③ 상기 방법을 반복한다.

추적이 잘 이루어지고 있을 때는 오차가 작으므로  $\alpha$ ,  $\beta$  값이 작게 평가될 수 있으나 실측위치 자체에도 오차가 있으므로  $\alpha$ ,  $\beta$ 를 1로 할 수는 없다. 목표물이 등속직선운동을 할 때에는  $\alpha$ ,  $\beta$ 를 작게해서 분산을 충분히 평활화 함으로써 높은 추적 유지 정밀도를 얻을 수 있다.

추적의 기본적인 동작 알고리즘은 다음과 같다.

- ① 현재까지 얻어진 각각의 물표마다 위치데이터를 상관된 트랙과 연결하여 메모리에 축적한다.
- ②  $\alpha$ ,  $\beta$  추적필터 방법으로 물표의 앞으로 위치를 예측한다.
- ③ 다음 예측 위치를 중심으로 추적 게이트(윈도우)를 설정한다.

추적 윈도우의 크기는 실제 레이다에서 통상적으로 4~5단계로 나누어 미리 설정하여 사용되고 있지만 본 논문에서는 10단계로 세분하여 설정하고 비교 테이블을 미리 작성하여 비교연산 동작이 빠르게 이루어지도록 하였다. 먼저 새로운 물표의 위치정보가 입력되면 최대게이트가 설정되고 연속되는 스캔 중에서 일정회수의 위치정보가 얻어지면 동일 목표로 간주하여 다음의 예측위치에서는 한 단계 작은 게이트가 설정된다. 그 게이트 내에 또다시 물표위치정보가 얻어지면 다음의 예측위치에서는 보다 작은 게이트가 설정된다. 이와 같이 해서 최종적으로는 최소게이트로 추적이 계속된다. 만약 설정된 게이트 내에 목표의 위치정보가 얻어지지 않으면 한 단계 큰 게이트로 넓어진다. 최대게이트로 확대해도 목표물의 위치정보가 얻어지지 않을 때는 손실목표로 간주하여 추적이 중지된다. 최소게이트는 추적 유지 기능을 손상시키지 않는 크기로 설정하여 접근하는 여러 목표물을 잘 분리할 수 있는 작은 크기가 되도록 상반되는 조건을 최대한 만족시키는 값으로 선택하였다.

(2) 물표 위치인식 알고리즘

항해용 레이다에서 TWS를 이용하여 물표를 추적하기 위해서는 물표의 위치 인식이 필수적인데, 이는 물표의 위치 결정에 따라 윈도우의 크기가 결정되기 때문에 매우 중요하다. 그러나 물표의 위치 인식 방법에 관해 공개된 기술은 거의 없으며, 이 기술은 하드웨어와 소프트웨어의 결합으로 이루어진다. 일반적으로 물표에서 반사되어 되돌아오는 물표신호의 세기를 몇 개 또는 몇 십 단계의 이산적인 양으로 나누어, 설정된 윈도우 내에서 반사되어 온 물표의 세기중심(또는, 「무게중심」)을 찾아 그 물표의 중심 위치로 인식하는 방법을 사용한다.

① 위치 인식에 필요한 정보들

물표의 위치를 인식하기 위해 레이다 시스템에서 받을 수 있는 정보는 물표 거리 정보, 방위각 정보, 반사된 물표(target echos)의 세기정보이며, 이들 정보로부터 물표의 무게 중심을 찾을 수 있다. 여기서 말하는 무게중심이란, 어떤 설정된 윈도우 내에서 반사된 물표 세기의 거리방향

중심("Rcc"라 표기함)과 방위각 방향 중심("Bcc"라 표기함)인 2차원상의 평면적 중심 위치를 말하는 것이며, 그것을 결정하는 요소는 위의 세가지 정보들이다.

물표의 위치 정보를 인식하기 위해서는 물표의 거리 및 방위각 정보가 디지털 값으로 변환되어 입력되어야 한다. 거리 정보는 일정한 기간 동안 샘플링 주파수에 의해 변환(Range Quantizing)되고, 방위각 정보는 Encoder나 Resolver, Synchro 등으로부터 받은 정보를 변환(Bearing Quantizing) 시킨다.

거리 정보의 분해능(resolution)은 반사된 물표에 대한 샘플링 주파수에 의해 결정되며, 샘플링 주파수가 fs일 때, 거리 양자화(Range Quantizing) 간격 ΔR은

$$\Delta R = 3 \times 10^8 / 2fs \text{ [m]}$$

이 되며, 샘플링 주파수가 증가하면 ΔR은 좁아지고 거리 분해능은 증가한다. 예를 들면 항해용 레이다에서 추적을 위해 사용되는 샘플링 주파수는 거리 범위에 따라 표 1과 같은 세가지의 다른 주파수가 이용될 수 있다.

<표 1> 거리 범위별 추적을 위한 샘플링 주파수

거리범위(NM)	0.75	1.5	3	6	12	24
샘플링 주파수 (MHz)	9.7192		4.8596			2.4298
샘플링 간격(R) (m)	15.5		31			62

한편 방위각 정보는 안테나 1회전시 받는 방위각 펄스의 수에 따라 틀러지는 데 일례로 1024개의 펄스를 받아 인식할 경우, 이 때의 방위각 양자화(Bearing Quantizing) 간격 ΔB는 ΔB = 360° / 1024 = 0.35° 로 주어진다.

② 수식적인 모델링

레이다에서의 물표에 대한 위치 인식은 레이다 안테나의 위치를 기준으로 한 극좌표계로서의 위치인식이다. 물표의 위치 인식을 위해 그 원도우 내에서 반사되어온 물표의 데이터를 고려하면 원도우에서 초기 방위각(Start Bearing) 과 초기 거리(Start Range)를 원점(B<sub>1</sub>, R<sub>1</sub>)으로 하여 거리('R'로 표기)를 가로축으로 하고, 방위각(B로 표시)을 세로축으로 하여 원도우 내에 들어온 물표 데이터를 표시할 수 있다. 거리는 일정 시간간격으로 샘플링되므로 R축은 일정한 거리 간격(ΔR)의 이산적인 양으로 표시되고, 방위각은 일정한 각도를 간격으로 한 디지털값으로 변환되므로 B축은 일정한 각도 간격(ΔB)의 이산적인 양으로 표시된다. 즉 B축과 R축은 모두 일정 간격의 이산적인 양으로 나뉘어지게 된다.

물표의 방위각은 B<sub>i</sub>, 거리는 R<sub>j</sub>인 물표의 반사파 세기를 X<sub>ij</sub>로 표시하고, 추적하려는 물표에 대한 원도우 크기는 B축 방향으로 m개, R축으로는 n개로 하여 반사되어 돌아온 물표의 세기 중심인 무게중심(B<sub>cc</sub>, R<sub>cc</sub>)을 찾는 수식은 다음과 식(26), (27)과 같다.

$$B_{cc} = \frac{\sum_{i=1}^m \left[ B_i * \sum_{j=1}^n X_{ij} \right]}{\sum_{i=1}^m \left[ \sum_{j=1}^n X_{ij} \right]} = \frac{\sum_{i=1}^m \sum_{j=1}^n B_i * X_{ij}}{\sum_{i=1}^m \sum_{j=1}^n X_{ij}} \quad (26)$$

$$R_{cc} = \frac{\sum_{j=1}^n \left[ R_j * \sum_{i=1}^m X_{ij} \right]}{\sum_{j=1}^n \left[ \sum_{i=1}^m X_{ij} \right]} = \frac{\sum_{j=1}^n \sum_{i=1}^m R_j * X_{ij}}{\sum_{j=1}^n \sum_{i=1}^m X_{ij}} \quad (27)$$

③ 무게중심 기법의 효과 및 구현 방법

물표의 무게중심으로 물표의 위치를 인식하게 되면 원도우 내에 들어오는 평균 잡음수준이나 반사파가 약한 것들에 의한 물표 반사파 형상변화에 영향을 덜 받게 된다. 따라서 약한 반사파의 변화에 의한 무게중심 위치 정보의 변화는 미약하여 방위각 및 거리 양자화 간격보다 작게 된다. 그러므로 무게중심 위치 인식의 분해능이 방위각 및 거리 양자화 간격보다 더 조밀하게 된다. 즉 샘플링 주파수를 증가시키지 않고 물표의 위치 정보 분해능을 증가시킬 수 있다.

일반적으로 레이다 스캐너인 안테나로부터 수신된 물표의 신호를 어떤 기준 수준과 비교하여 물표의 유무 상태만을 탐지하여 물표의 위치를 탐지하게 되면, 방위각 및 거리 양자화 간격에 의해 생기는 물표 위치 인식상의 오차(Quantization Error)는 피할 수 없다. 이것을 줄이기 위해서는 샘플링 주파수를 증가시켜 양자화 간격을 줄여야 한다. 반면에 물표의 세기를 변수(X<sub>ij</sub>)로 받고, 그 변수의 무게 중심으로 위치를 인식하게 되면 샘플링 간격을 줄이지 않고 물표의 위치정보의 분해능을 증가시키는 효과를 얻을 수 있으므로 양자화 오차에 대한 보상 효과를 얻을 수 있다.

결과적으로 TWS 레이다에서 물표의 위치 인식은 추적을 위해 필요한 것인데, 무게중심으로 물표의 위치를 인식하게 되면 위치정보의 분해능을 증가시키는 효과에 의해 보다 정확한 위치 인식이 가능하게 된다. 따라서, 동일 조건의 추적필터에 이와 같이 무게중심 기법을 적용한다면 보다 우수한 추적기능을 가지게 된다.

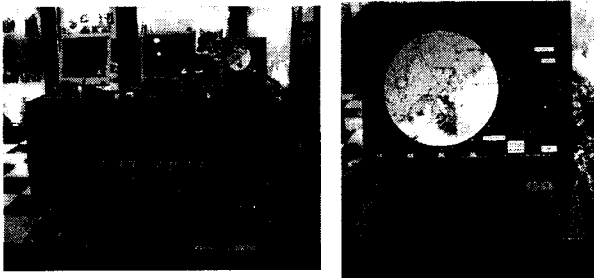
위에서 검토된 무게중심을 찾는 개념을 실제로 구현하는 문제에 있어서 곱셈과 덧셈, 나눗셈의 연산 횟수가 상당히 증가하며 그 연산을 전적으로 소프트웨어로만 처리하게 되면 신호처리를 위한 PC 연산시간이 많이 걸리게 되고 또한 원도우내의 모든 데이터를 저장할 기억장치의 공간도 필요하게 된다. 따라서 실제 구현상에 있어서 무게중심 알고리즘의 구현은 소프트웨어뿐만 아니라 하드웨어의 뒷받침이 있어야 구현할 수 있다.

4. 결론



해상인명안전협약(SOLAS)은 2002년 7월 1일부터 신조되는 총톤수 500톤 이상의 모든 선박에 선박 자동추적이 가능한 ATA를 탑재시키도록 되어있으나 기존에 운항중인 10,000톤 미만의 중소형 선박은 탑재할 의무가 없으므로 특히 무중시 타선의 움직임을 정확히 파악하지 못하여 발생하는 충돌사고가 빈번한 실정이다.

따라서 본 논문에서는 소형선박 레이더의 표시장치에서 나타나는 다양한 물표 혹은 목표물의 레이더 신호를 RSC로 변환, 입력받은 신호를 처리하여 주위에 운항하고 있는 선박들과 각종 물표들을 자동으로 축출 인식하고 상대선의 움직임을 자동 추적할 수 있는 기능을 가진 ATA를 연구 개발함으로써 선박의 안전운항을 확보함에 있다. ATA에서 가장 핵심적인 기술인 목표물 자동추적 알고리즘을 연구 개발하여 실제 장비에 적용 제작하고, 국제표준(IEC60872-2:1999) 및 MSC.64 (67) Annex4의 시스템 시험항목과 검증에 따라 그 실효성을 확인해 본 결과 성능기준을 만족하였다.



<그림 8> 마린위크 2003에 전시중인 ATA의 시작품

그러므로 제안된 자동추적 알고리즘을 적용한 ATA를 연안선의 중소형 선박에 저렴한 가격으로 보급함으로써 연안 충돌사고 및 좌초를 미연에 방지하는 기능의 확보로 해상안전을 도모할 수 있을 것이다. 또한 선교 항해당직 경보시스템(BNWS: Bridge Navigational Watch Alarm System)과 연결하여 경보 기능을 강화시키고, 전자해도(ENC)를 가볍게 사용하면서 갱신 데이터의 활용이 가능한 간이전자해도(ECS) 데이터베이스로 변환하여 이 ATA와 접목시킴으로써 선박의 해상 안전운항을 더욱더 확고히 할 수 있을 것으로 생각된다.

### 참고문헌

- [1] 안조영, "기동 표적 추적을 위한 준최적 칼만 필터 설계", 한국과학기술원 석사학위논문, 1986.
- [2] Yaakov Bar-Shalom, Xiao-Rong Li, "Multitarget - Multisensor Tracking: Principles and Techniques", YBS, 1995.
- [3] Yaakov Bar-Shalom, "Multitarget - Multisensor Tracking: Advanced Applications", Artech House, 1990.
- [4] 구자윤 외 5명, "RADAR · ARPA의 이론과 실무", 한국해양수산연수원, 1998.
- [5] 양규식, "RADAR 이론과 보수", 아성출판사, 1986.
- [6] 구자윤, 김석재 외 8명, "전자해도 이용 선박자동추적장치 (ATA) 개발용역-1차년도 보고서", 해양수산부, 2002.12.