

# An Application of Deep Clustering for Abnormal Vessel Trajectory Detection

Heon-Jei Park\* · Jun Woo Lee\*\* · Ji Hoon Kyung\* · Kyeongtaek Kim\*<sup>†</sup>

\*Department of Industrial Engineering, Hannam University

\*\*GDL System

## 딥 클러스터링을 이용한 비정상 선박 궤적 식별

박헌제\* · 이준우\*\* · 경지훈\* · 김경택\*<sup>†</sup>

\*한남대학교 산업공학과

\*\*㈜지디엘시스템

Maritime monitoring requirements have been beyond human operators capabilities due to the broadness of the coverage area and the variety of monitoring activities, e.g. illegal migration, or security threats by foreign warships. Abnormal vessel movement can be defined as an unreasonable movement deviation from the usual trajectory, speed, or other traffic parameters. Detection of the abnormal vessel movement requires the operators not only to pay short-term attention but also to have long-term trajectory trace ability. Recent advances in deep learning have shown the potential of deep learning techniques to discover hidden and more complex relations that often lie in low dimensional latent spaces. In this paper, we propose a deep autoencoder-based clustering model for automatic detection of vessel movement anomaly to assist monitoring operators to take actions on the vessel for more investigation. We first generate gridded trajectory images by mapping the raw vessel trajectories into two dimensional matrix. Based on the gridded image input, we test the proposed model along with the other deep autoencoder-based models for the abnormal trajectory data generated through rotation and speed variation from normal trajectories. We show that the proposed model improves detection accuracy for the generated abnormal trajectories compared to the other models.

**Keywords :** Deep Learning, Anomaly Detection, Vessel Trajectory

### 1. 서 론

2000년 6월 국제해사기구(선박 종류별로 기준을 초과하는 선박에 대하여 AIS(선박자동식별장치, Auto Identification System) 장비 장착을 의무화하였다. 국내에서는 2008년 선박안전법이 개정된 이래, 연해구역을 항해하는 선박은 선종별 기준에 따라 AIS 신호 발신 장비를 탑재하

고 있다. 수집된 AIS 데이터는 해상 교통 관련 연구에서 선박 이동의 패턴을 모형화 할 때 신뢰성 있는 입력 자료로 사용되면서 연구가 활발해 지고 있다. 이는 GPS 장비가 보편화 되면서, 육상의 차량 이동, 보행자 이동, 동물의 이동 등을 나타내는 궤적을 이용한 이동 패턴에 관한 연구가 활성화된 것과 유사한 현상이다. 선박의 해상통로는 근사적으로 정해져 있고, 안전상의 이유로 주위의 다른 배들과 멀리 떨어져 운항해야 하므로, 도로에 국한된 움직임을 보이는 육상의 차량 궤적과는 달리, 선박은 비교적 자유롭게 움직인다. 따라서, 두 선박의 궤적이 비슷한 패턴을 나타내어도, 두 궤적간의 유사도는 낮을 수 있다[6].

Received 18 November 2021; Finally Revised 12 December 2021;  
Accepted 20 December 2021

<sup>†</sup> Corresponding Author : kim610@gmail.com

이와 같은 이유로 궤적에 대한 모양(shape) 유사도만을 계산하는 전통적인 궤적 유사도 기반 척도 방식인 동적시간 워핑(Dynamic Time Warping), 최장공통부분순서(Longest Common Subsequence), 편집거리순서(Edit Distance on Real sequence) 등은 선박 궤적 연구에 직접 적용하는데 한계가 있다[6].

최근 딥러닝의 발달로, 고차원의 공간에서 표현되는 궤적을 비선형변환을 통하여 특성(feature)을 해치지 않으면서 저차원 잠재(latent) 공간의 데이터로 전환이 가능해졌다. 저차원으로 표현된 특성치를 얻은 후, 군집화가 쉽도록 특성치의 표현을 바꾼 다음 클러스터링을 하는 방법을 딥 클러스터링(Deep Clustering)이라 부른다[8, 12, 16].

2020년에 중국으로부터의 밀입국 선박을 탐지하지 못하여 밀입국자들이 서해안에 상륙했다는 뉴스가 큰 주목을 받았다. 또한 2019년에는 북한의 소형선박이 동해안 항구로 들어 왔어도 탐지하지 못하는 사고도 있었다. 그 원인으로 경계태세의 부족이 지적되었으나, 짧은 시간의 선박 궤적 탐지로는 비정상 궤적 여부를 판단할 수 없다.

본 논문에서는 AIS 데이터를 입력으로 받아 딥 클러스터링을 이용하여 선박의 비정상 궤적을 자동 식별하는 모형을 제안한다. 또한, 제안 모형의 성능을 검증하기 위하여, 비정상 궤적을 생성하고, 이를 기존 오토인코더(Autoencoder) 모형과 딥 클러스터링 모형에 각각 적용하여 식별률을 상호 비교하는 실험도 수행하였다. 본 논문은 비정상 선박 궤적 식별에 딥 클러스터링을 적용한 최초의 논문이며, 대부분의 관련연구에서 식별사례만을 제시한 연구와는 달리, 다양한 방법으로 비정상 궤적을 생성하여 식별률을 제시한 최초의 논문이다. 본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 요약하고, 3장에서는 필요한 이론적 기초를 설명한다. 4장에서는 본 논문에서 제시하는 모형을 소개하고, 5장에서는 실험 내용 및 결과를 요약하고 분석하며, 6장에서 결론을 맺는다.

## 2. 관련연구

AIS 데이터와 오토인코더를 이용한 비정상 선박 궤적 식별에 관한 연구는 다음과 같다. Li et al.[9]은 위도 및 경도를 나타내는 궤적 이미지를 오토인코더를 이용하여 차원을 축소한 후 클러스터링을 하는 방법이, 기존의 거리 기반 클러스터링 방법보다 성능이 뛰어남을 보였다. 궤적 유사도 기반 클러스터링에 목적을 둔 연구로 비정상 궤적 선박 식별에 응용 가능할 것으로 보인다. Liang et al.[11]은 선박 AIS 데이터의 궤적을 이미지화할 때, 위도와 경도로 셀의 위치를 정하고, 셀의 면적 내에서 보고된 AIS 신호의 횡수로 셀의 값을 정하는 방법을 사용하였다. 합성곱

오토인코더를 이용하여 궤적 이미지의 차원을 축소한 후 클러스터링을 하는 방법이, 기존의 거리 기반 클러스터링 방법보다 성능이 뛰어남을 여러 가지 성능 척도를 사용하여 보여 주었다. 이 연구는 비정상 궤적 탐지 연구로의 확장이 가능하다. Oh et al.[14, 15]은 3분간의 항적 데이터를 입력 받아 오토인코더로 차원을 축소한 후, 오토인코더 복원 에러, 클러스터링 에러, 전역 이상치 등을 사용하여 비정상 궤적 선박을 식별하는 시스템을 제안하였다. 이 시스템은 3분 단위의 데이터를 사용하므로, 실시간 관제에 도움이 되는 장점이 있다. 이상 궤적의 식별 사례는 자세히 설명하였으나, 비정상 궤적의 식별률을 제시하기 위한 실험은 수행하지 않았다. 최근에 Park et al.[17]은 경도, 위도, 속도, 방향, 시간을 각각 원-핫 인코딩(One-Hot Encoding) 형태로 전처리를 한 후에 변분 오토인코더(Variational Autoencoder)와 관리도를 결합하여 비정상 궤적을 식별하였다. AIS 데이터를 이용한 비정상 궤적 선박 탐지의 연구 대부분은 비정상 궤적의 탐지 사례를 제시할 뿐, 제안한 모형이 비정상 궤적을 어느 정도 탐지하는지에 대한 실험은 수행하지 않았다.

이와는 달리, Son et al.[20]은 비정상 궤적의 식별률을 최초로 제시 하였다. 선박 AIS 데이터 중 위도 및 경도를 이용하여 궤적을 이미지로 만든 후 오토인코더를 이용하여 이상거동 선박을 식별하는 시스템을 제안하고, 제안된 시스템을 이용하여 기존 궤적을 동서남북 방향으로 평행 이동하여 비정상 궤적을 생성한 후, 이들 비정상 궤적을 식별하는 실험을 하였다. 그러나, 평행 이동한 비정상 궤적은 입출항 항구 부근에서도 평행 이동된 채로 표현되어, 발생할 가능성이 희박한 궤적으로 실험한 단점이 있다.

Xie et al.[23]은 완전연결(Dense) 레이어를 이용하여 오토인코더를 구성하고, 단계 2에서 자기 지도학습 방식을 사용하여 클러스터링 손실을 최적화한 딥 클러스터링 시스템인 Deep Embedded Clustering(DEC) 모형을 제시하였다. Guo et al.[7]은 DEC 모형이 잠재공간을 오염시킬 수 있다고 주장하며, 단계 1과 단계 2를 통합하여 최적화한 모형을 제시하였다. Guo et al.[8]은 완전연결 오토인코더 대신에 합성곱(Convolutional) 오토인코더를 사용하여 정확도를 향상한 Deep Convolutional Embedded Clustering(DCEC) 모형을 제안하였다. Aytakin et al.[3]은 오토인코더에  $l_2$  정규화를 포함하는 네트워크가 다른 모형들보다 클러스터링 정확도가 더 높음을 보였다. 이러한 모형들은 MNIST 데이터 셋 및 USPS 데이터 셋에 대한 클러스터링 정확도를 계산하여 모형의 우수성을 제시하고 있을 뿐, 실제 데이터에 적용한 사례는 매우 드물다.

궤적이 제한되는 육상 교통 분야와는 달리, 항공 분야와 해상 분야는 궤적이 비교적 자유롭다는 공통점이 있다. Oliver et al.[16]은 딥 클러스터링을 이용하여 항공기의 이상

궤적을 식별하였다. ADS-B(Automatic Dependent Surveillance - Broadcast) 데이터를 이용하여 저자들이 제시한 artefact[16]와 DCEC 모형[8]을 실험하였으나, 정확도의 비교는 하지 않았다

### 3. 배경이론

#### 3.1 클러스터링

클러스터링은 유사도 함수 또는 거리 함수를 사용하여 비슷한 데이터들을 군집화하는 비지도학습(unsupervised learning) 기반의 데이터 분석 방법이다. 점(point)들을 클러스터링하는 경우는 두 점 사이의 거리를 명확히 정의할 수 있으므로 문제가 없다. 궤적들을 클러스터링하는 경우에는 두 궤적의 길이가 다를 수 있으므로, 점들을 클러스터링할 때 사용하는 거리 함수를 그대로 적용할 수 없다. 따라서, 궤적을 클러스터링하기 전에, 같은 길이가 되도록 궤적을 변환하는 전처리(pre-processing)한다. 전처리 방법은 크게 선형변환 방법과 비선형 변환 방법으로 나눌 수 있다[4, 12]. 선형 변환 방법으로는 궤적에 주성분 분석을 적용하여, 일정한 수의 주성분 계수를 추출하여 궤적을 나타내도록 하는 방법[24]과, 시계열 자료를 이산 푸리에 변환을 통하여 주파수 영역의 데이터로 변환하여, 시계열 데이터의 전체적인 이미지를 나타내는 일정한 수의 저주파 영역 값으로 궤적을 나타내도록 하는[13] 방법 등이 있다. 비선형 변환 방법으로는, 궤적 데이터를 피셔커널(Fisher kernel)을 이용하여 커널로 사용한 모형의 파라미터 수에 의해 크기 결정되는 고정된 벡터로 변환하는 방법[2]과, 데이터를 특성 벡터로 나타낸 후 공분산(covariance) 벡터를 구하여 같은 크기로 나타내는 방법[10] 등이 있다.

그럼에도 불구하고, 이러한 변환을 통하여 궤적의 크기를 동일하게 만든 후에 대응하는 두 점 사이의 거리를 기반으로 진행되는 클러스터링은, 궤적의 일정하지 않은 샘플링 레이트(sampling rate)로 궤적 데이터를 얻었거나, 낮은 샘플링 레이트로 궤적 데이터를 얻었거나, 노이즈가 포함된 궤적 데이터인 경우 등, 궤적 데이터의 구조에 영향을 주는 요소로 인하여, 대응하는 점을 어떻게 결정 하는가에 따라 성능이 변하며, 노이즈의 포함 여부가 성능에 영향을 준다[21].

#### 3.2 딥 클러스터링

딥 클러스터링은 딥러닝을 이용하여 주어진 데이터를 군집화가 용이한 형태를 갖는 저차원 데이터로 변환하여 클러스터링 수행한다. 딥 클러스터링은 사용하는 인공신

경망의 종류에 따라 다시 나누어지는데, 본 논문에서는 오토인코더를 사용하므로, 이를 중심으로 딥 클러스터링을 살펴본다. 딥 클러스터링의 목적 함수는 손실(loss)로 나타낸다. 손실  $L$ 은 일반적으로 네트워크 손실  $L_r$ 과 클러스터링 손실  $L_c$ 로 구성되며, 다음과 같이 정의한다.

$$L = L_r + \lambda L_c \quad (1)$$

여기에서  $\lambda$ 는 클러스터링 손실의 가중치를 조절하는 하이퍼 파라미터를 나타낸다. 네트워크 손실  $L_r$ 은 특성을 학습하고 잠재공간이 오염되는 것을 막는다. 클러스터링 손실  $L_c$ 은 잠재공간에서 차원이 축소되어 표현된 궤적들이 클러스터를 잘 형성하도록 촉진한다. 사용하는 인공신경망의 종류에 따라 네트워크 손실은 다르게 표현되는데, 오토인코더를 사용하는 경우에는 복원 손실(reconstruction loss)이 네트워크 손실에 해당하며, 변분 오토인코더를 사용하면 변분 손실이, 그리고 생성적 적대 신경망을 사용하면 적대적 손실(adversarial loss)이 네트워크 손실에 해당한다.

오토인코더는 비지도학습에 사용되는 인공신경망의 하나로, 입력 데이터의 차원을 축소하는 인코더(encoder)와 축소된 데이터를 원래의 데이터로 복원하는 디코더(decoder)로 구성된다. 오토인코더를 이용한 딥 클러스터링은 다음 두 단계로 네트워크를 훈련시킨다.

단계 1: 복원 손실만을 이용하여 오토인코더를 훈련한다. 차원이 축소되어 잠재 특성(latent feature)을 가진 벡터를 추출한다.

단계 2: 목표 분포(target distribution)를 계산한 다음 클러스터링 손실을 최소화하는 절차를 반복 실행하여 최적화한다.

### 4. 비정상 궤적 식별 모형

#### 4.1 입력 이미지 생성

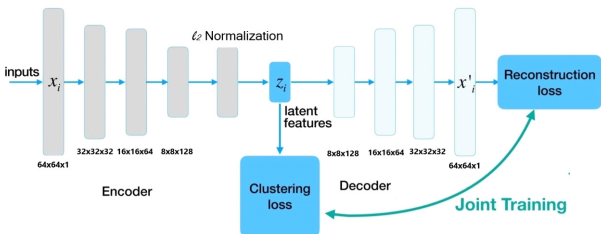
선박의 궤적은 선박의 위치를 시간 순으로 배열한 시계열 데이터다. 인공신경망을 이용하여 궤적을 분류하는 작업은 시계열 데이터의 길이가 일정하지 않기 때문에 쉽지 않다[19]. 만일 궤적 길이의 변화가 크면, 전처리 과정이 필수적이다[19]. 본 논문에서는 궤적의 길이에 상관없이 하나의 이미지로 변환함으로써 이 문제를 해결한다.

주어진 AIS 데이터 셋을 제안한 딥 클러스터링 모형의 입력으로 사용하기 위하여  $64 \times 64$  크기의 이미지(즉 행렬)로 변환하는 전처리 과정을 수행한다. 먼저, 주어진 AIS

데이터 셋을 선박식별번호별, 날짜 별, 시간 순으로, 위도, 경도 및 속도를 배열한다. 그 다음, 위도의 최소값과 최대값의 차이와 경도의 최소값과 최대값의 차이를 이용하여 선박의 궤적의 위치를  $64 \times 64$  셀에서 행의 위치와 열의 위치로 나타낸다. 셀의 값은 평균속도를 나타낸다.

## 4.2 모형의네트워크 구조

제안된 모형은 전체적으로 <Figure 1>과 같이 오토인코더를 사용하는 딥 클러스터링의 네트워크 구조를 따른다. 인코더의 각 합성곱레이어의 필터의 크기와 수는 각각  $32 \times 32$ ,  $16 \times 16$ ,  $8 \times 8$ 과  $32$ ,  $64$ ,  $128$ 로 설정되어 있으며, 활성화 함수로 Relu 함수를 사용한다.  $2 \times 2$  그리드의 최대 풀링 레이어가 각 합성곱 레이어 다음에 위치한다. 합성곱 레이어와 풀링 레이어를 모두 통과하면 데이터의 크기는  $8 \times 8 \times 128 = 8192$ 가 된다. 이후, 완전연결 레이어를 통하여 데이터는 10으로 축소된 후,  $l_2$  정규화를 실행한다.  $l_2$  정규화가 된 데이터는 디코더에 입력되어, 데이터의 크기가 8,192로 확대된 다음, 업 샘플링 레이어와 합성곱 레이어를 통하여 원래의 이미지의 크기로 복원된다. 본 논문에서 제안된 이 모형을 DCECL(Deep Convolutional Embedded Clustering with  $l_2$  normalization) 모형으로 부르기로 한다.



<Figure 1> Architecture of Deep Clustering based on Autoencoder (Adapted from FIGURE 1 in [12])

## 4.3 모형의훈련

제안한 DCECL 모형에서 입력을  $x_i$ 라 하고, 출력을  $x'_i$ 라 하자.  $f_k$ 를 인코더 레이어  $k$ 가 레이어 입력을 레이어 출력으로 변환시키는 함수라 하자. 그러면 인코더 마지막에서  $l_2$  정규화를 하는 레이어를 제외하고  $n$ 개의 레이어를 갖는 인코더의 출력  $w_i$ 는 다음과 같다.

$$w_i = f_n(f_{n-1}(\cdots f_1(x_i))) \quad (2)$$

인코더의 마지막에  $l_2$  정규화를 수행하므로, 인코더 출

력  $z_i$ 는 다음과 같다.

$$z_i = \frac{w_i}{\|w_i\|_2} \quad (3)$$

디코더의 레이어  $p$ 의 입출력 변환 함수를  $g_p$ 라 하면,  $m$ 개의 레이어를 갖는 디코더의 출력  $x'_i$ 는 다음과 같다.

$$x'_i = g_m(g_{m-1}(\cdots g_1(z_i))) \quad (4)$$

오토인코더의 입력 이미지와 출력 이미지를 비교하여 그 차이를 복원 손실로 나타낸다. 평균 제곱 오차(Mean Squared Error)를 손실 함수로 사용하면 복원손실  $L_r$ 은 다음과 같다.

$$L_r = \frac{1}{|I|} \sum_{i \in I} (x_i - x'_i)^2 \quad (5)$$

인코더의 출력  $z_i$ 은 입력을  $x_i$ 를 비선형 변환을 하여 차원이 축소된 형태이며,  $l_2$  정규화된 데이터이다. 인코더의 출력을 이용하여 클러스터링을 한다.  $l_2$  정규화된 데이터는 단위 벡터로 볼 수 있으므로, 두 벡터 간의 거리는 코사인 거리(cosine distance)로 정의된다[5].

클러스터링 손실을 정의하기 위하여, DEC[23] 및 DCEC[8]에서 사용한 soft assignment 개념을 이용하여  $Q$  분포를 정의한다.  $K$ 개의 클러스터의 중심  $\{\mu_j\}_1^K$ 와 DCECL 모형의 인코더 출력  $z_i$ 에 의해 표현되는  $Q$  분포는 다음과 같이 정의한다.

$$q_{ij} = \frac{(1 + d(z_i, \mu_j))^{2^{-1}}}{\sum_j (1 + d(z_i, \mu_j))^2} \quad (6)$$

$$d(z_i, \mu_j) = 1 - \frac{z_i \cdot \mu_j}{\|z_i\| \|\mu_j\|} \quad (7)$$

$P$ 는 목표 분포로 다음과 같이 정의된다.

$$p_{ij} = \frac{q_{ij}^2 / \sum_i q_{ij}}{\sum_j (q_{ij}^2 / \sum_i q_{ij})} \quad (8)$$

$Q$  분포를 제공한 후 정규화함으로써, 목표 분포  $P$ 는 soft assignment가 0 또는 1로 가까이 가도록 강제한다[16, 23].

클러스터링 손실은  $P$  분포와  $Q$  분포를 이용하여 다음과 같이 쿨백-라이블러(Kullback-Leibler) 거리로 나타낸다.

$$L_c = KL(P \| Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \quad (9)$$

## 5. 실험

본 논문에서 제안한 DCECL 모형의 성능을 나타내기 위하여, 기존 연구에서 제시된 DCEC 모형[8], AEL 모형, 그리고 기본 오토인코더(AE) 모형과 비교하였다. DCEC 모형에서 오토인코더 부분만을 떼어내어 AE 모형의 네트워크로 구성하였으며, AE 모형의 인코더 마지막에  $l_2$  정규화 레이어를 추가하여 AEL 모형의 네트워크로 구성하였다. 따라서, 네 모형 모두 네트워크 구조에 있어 공통되는 오토인코더 부분은 레이어 수와 필터의 크기 및 수, 활성화 함수, 풀링 레이어의 크기가 동일하며, 자세한 내용은 3.2 절의 DCECL 모형의 네트워크 구조에 나타나 있다.

### 5.1 데이터전처리 및 사전 훈련

모형들을 훈련하기 위하여, 1개월간 평택항으로 입출항하는 선박들의 AIS 데이터를 이용하여, 3.1절에서 제시된 방법으로  $64 \times 4$  크기의 이미지로 변환하였다. 네트워크 훈련은, 2.2절의 단계 1에서 epoch는 2,000으로 설정하였으며, 단계 2에서도 epoch를 2,000으로 설정하였다. 손실 함수의 가중치는 복원 손실의 크기와 클러스터링 손실의 크기를 고려하여  $\lambda = 0.01$ 로 하였다.

### 5.2 비정상궤적 생성

제안한 모형의 성능을 테스트하기 위하여, 훈련에 사용한 입력 이미지로부터 궤적의 크기가 40셀 - 100셀인 궤적 중에서 선박의 유형별 궤적 수에 비례하여 무작위로 200개를 선택한다. 선택된 궤적에 대하여, 항구에서 가까운 순서대로 10개의 궤적 셀은 그대로 둔 채 나머지 궤적 셀의 속도 값을 변경하거나 궤적의 위치를 변경하여, 다음과 같이 두 종류의 비정상 궤적을 만든다.

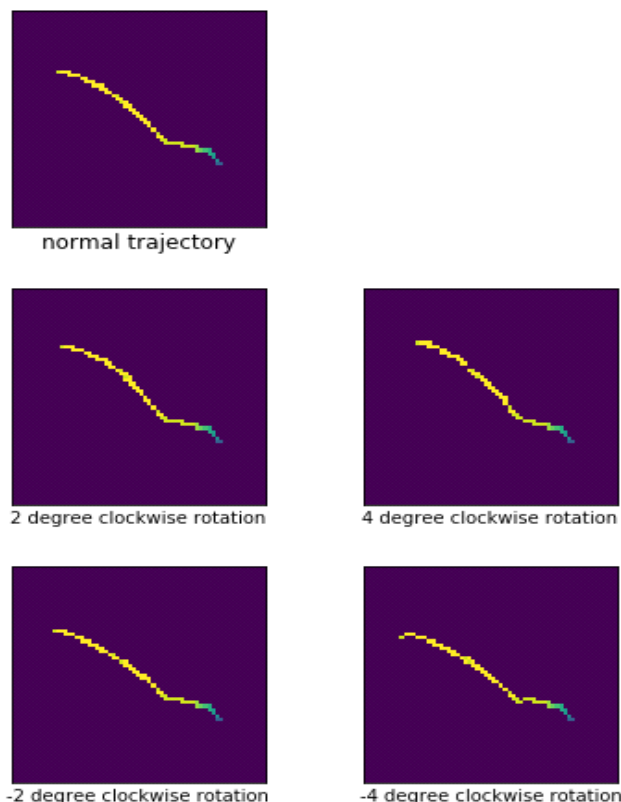
#### 5.2.1 랜덤속도 비정상 궤적

선택된 궤적의 선박 유형과 같은 유형의 선박들에 대하여, 항구에 가까운 10개 궤적 셀을 제외한 나머지 궤적 셀의 평균 속도와 표준편차를 구한다. 평균 속도의 0.4배에서 1.6배까지 0.2씩 증가시키면서 7개의 테스트 셋을 만든다. 각 테스트 셋은 200개의 궤적으로 이루어져 있으며, 주어진 평균속도 배율과 선박 유형별 평균속도 및 표준편차를 이용하여, 정규 분포에서 임의로 선택하여 10개의 궤적 셀을 제외한 나머지 셀들의 속도 값을 배정한다.

#### 5.2.2 시계 방향/반시계 방향 비정상 궤적

선택된 선박의 궤적을 이루는 셀 중에서 항구에 가까운 10개 궤적 셀은 그대로 두고, 10번째 궤적 셀을 기준점으

로 삼아, 나머지 셀들의 위치를 시계방향으로 1도, 2도, 3도, 4도, 5도 및 -1도, -2도, -3도, -4도, -5도 이동하여, 10개의 데이터 셋을 만든다. 각 데이터 셋은 200개의 궤적으로 이루어져 있다. 이때, 셀의 속도 값은 회전하기 전의 속도 값을 그대로 유지한다. <Figure 2>는 정상 궤적과 회전된 비정상 궤적의 예를 보여 준다.



<Figure 2> Abnormal Trajectory Images Generated by Rotation of the Normal Trajectory

### 5.3 사전 훈련과 클러스터 수 및 임계값 설정

궤적의 비정상 여부를 정의할 때 사용하는 임계값을 설정하는 문제는 논란의 대상이다[16]. 모두가 동의하는 최적 임계값은 없으므로, 문제마다 여건에 맞게 설정해야 한다. 실제 상황에서는 비정상 궤적을 정의할 때 사용하는 임계값을, 주어진 해상 감시 영역의 크기와 해상 감시 자원 등을 고려하여 결정하여야 한다. 본 실험에서는 복원 손실의 경우 (평균+3시그마)를 임계값으로 사용하기로 한다. 먼저 비정상 궤적을 가진 데이터가 훈련데이터에 섞여 있는 것을 방지하기 위하여, 훈련데이터를 입력 받아 복원 손실만을 이용하여 오토인코더를 훈련하고, 복원 손실이 임계값 이상인 훈련데이터는 제외하였다. 정상 궤적만으로 구성된 훈련데이터를 이용하여 새롭게 오토인코더를

훈련한 후, 임계값 이하인 데이터에 대하여 DCEC, DCECL 모형을 적용한 다음, elbow 방법을 적용하여 최적 클러스터 수를 도출하였다.

복원 손실이 임계값 이하인 궤적을 이용하여 클러스터링 임계값을 결정한다. 주어진 궤적은 각 모형에서 제시하는 방법에 따라 클러스터에 속할 확률이 가장 큰 클러스터에 배정된다. 각 클러스터에 배정된 궤적들에 대하여 그 클러스터 중심까지의 거리를 순서대로 정렬하여, 하위 2%에 해당하는 값을 그 클러스터의 임계값으로 사용하였다.

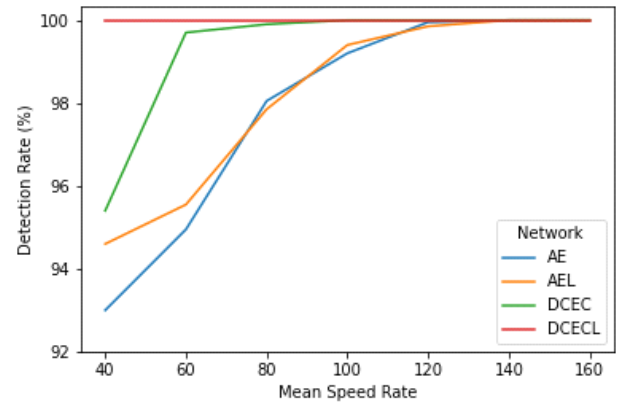
이 임계값들은 반복 실험을 시행할 때마다, 네트워크의 파라미터 값이 바뀌므로, 복원 손실의 분포도 바뀌며 각 클러스터 중심까지의 거리가 바뀌므로, 복원 임계값 및 클러스터링 임계값도 새롭게 자동 계산된다.

#### 5.4 비정상궤적 식별 결과 및 분석

실험에 사용한 모형 모두 평균속도보다 높은 속도 구간에서의 식별률이 매우 높았다. 그러나, 선박의 속도가 평균속도보다 낮아질수록 식별률이 조금씩 낮아졌으며, 40% 정도의 속도로 운항하는 경우는 식별률이 상대적으로 저조하였다. (평균+3시그마)를 임계값으로 사용한 경우, <Figure 3>에서 알 수 있듯이 비교 대상이 되는 세 모형에 비하여 DCECL 모형의 식별률이 훨씬 우수하였다. (평균+2시그마)를 임계값으로 사용하여도 유사한 경향을 보였다. DCECL 모형의 식별률이 우수한 이유는, 유사하지 않은 데이터가 인코더의 마지막 레이어의 유클리드 공간에서 출력 벡터 형태로 나타날 때, 유사하지 않은 속성을 정확하게 유지하지 못하기 때문에 구형(spherical) 또는 쌍곡 임베딩(hyperbolic embedding) 공간에서 출력 벡터를 나타내는 것이 낫다는 연구[22]와, MRI 이미지로부터 뇌 병변을 식별하는 실험에서  $l_2$  정규화를 사용하는 경우, 사용하지 않은 모형의 식별률보다 3% - 11% 높았다는 연구[18]도 본 연구의 결과와 맥락을 같이 한다.

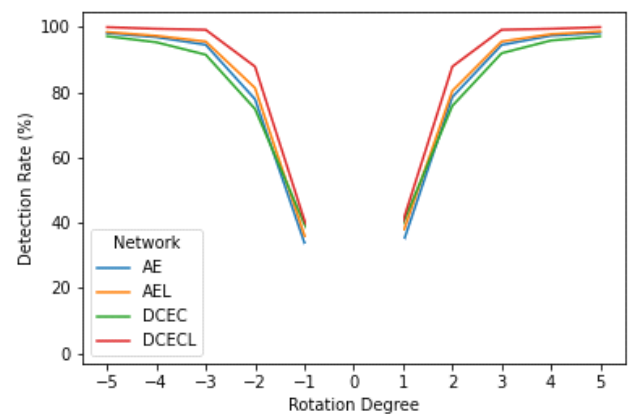
정상 속도 또는 그 이상으로 평균속도를 설정한 비정상 궤적의 경우, 비정상 궤적으로 식별하는 비율이 네 모형 모두 상당히 높았다. 본 연구에서 입력으로 사용한 이미지 데이터는 속도와 공간을 동시에 표현하고 있다. 일반적으로 시공간 데이터를 다룰 때, 각 데이터가 서로 독립이라는 일반적인 가정이 성립하지 않는다[1]. 이와 마찬가지로, 본 논문에서 사용한 속도-공간데이터도 서로 독립적이지 않다고 추정할 수 있다. 그런 이유로, 동일한 평균과 분산을 사용하여도 속도를 임의로 재배열하면 속도 데이터와 공간 데이터가 독립성을 가지므로, 오토인코더는 이를 비정상 궤적으로 탐지한다고 설명할 수 있다. 속도가 느린 구간에서 상대적으로 식별률이 낮아지는 경향이 있는데, 이는 궤적이 아닌 셀에서의 속도가 0으로 표현되므로, 속

도가 느릴수록 궤적이 아닌 셀과의 속도 차이가 작아지기 때문일 수 있다고 추정된다.



<Figure 3> Relationship among Detection Rate, Mean Speed Rate, and Network for 3 Sigma Threshold

회전 비정상 궤적을 사용하는 경우, <Figure 4>에서 알 수 있듯이, 시계 방향/반시계 방향으로 회전 각도가 커질수록 식별률이 높아지는 경향을 보였다. <Figure 4>에서 회전각 0도는 정상 항로를 나타내므로 탐지율을 나타내지 않았다. 회전된 비정상 궤적의 경우, 본 논문에서 제시한 DCECL 모형의 식별률이 다른 모형에 비하여 약간 높았다.



<Figure 4> Relationship among Detection Rate, Rotation Degree, and Network for 3 Sigma Threshold

DCECL 모형의 경우, (평균+3시그마)를 임계값으로 사용했을 때의 재구성 손실로 인한 식별률과 하위 2%를 임계값으로 사용했을 때 클러스터링 손실로 인한 식별률은 <Table 1>과 같다.

AIS 데이터 중에서 시간에 따른 선박의 위치 및 속도를 이미지화한 후 입력하면, 오토인코더의 복원 임계값으로



부터 대다수의 비정상 궤적을 식별한다는 특징을 가진다. 회전 각도가 1도 또는 -1도인 비정상 궤적 시뮬레이션 데이터에 대하여서는, 복원 오류로 인한 식별률은 낮았다. DCECL 모형의 경우, 클러스터링 임계값에 따른 비정상 궤적 식별률은 4.6%였으며, 이보다 높은 회전 각도의 경우, 식별률이 미미하였으나, 이미 높은 식별률에서 추가로 증가한 식별률이란 점에서는 미미한 증가도 의미가 있다.

<Table 1> Detection Rate (%) of Loss Source in DCECL for Abnormal Trajectories Generated by Rotation

Rotation Degree	Reconstruction	Clustering	Total
-5	100	0	100
-4	99.4	0.1	99.5
-3	99.1	0.1	99.2
-2	86.5	1.4	87.9
-1	36.3	4.6	40.8
1	35.9	4.6	50.5
2	86.5	1.4	87.9
3	99.1	0.1	99.2
4	99.5	0	99.5
5	100	0	100

오토인코더를 이용하는 딥 클러스터링 모형은 복원 손실과 클러스터링 손실을 최소화하도록 훈련을 하지만 성능은 오직 클러스터링 오류에 의해 결정된다. 한편, 본 논문에서 다루는 이상 식별 문제에서는 복원 임계값에 의한 식별률과 복원 오류를 일으키지 않은 데이터에 대한 클러스터링 임계값에 의한 식별률의 합으로 성능이 정의된다. 따라서, 딥 클러스터링에서 높은 성능을 가진 모형이 비정상 궤적 식별 문제에서도 항상 높은 성능을 가질 수는 없다. 딥 클러스터링 문제에서 뛰어난 성능을 보인 AEL 모형이, 보다 성능이 낮은 것으로 알려진 AE 모형에 비하여 본 논문의 실험에서는 우위를 보이지 못한 결과를 가져온 이유가 이 때문이다.

## 6. 결 론

본 논문에서는 선박의 AIS 데이터 중에서 선박식별번호, 위도, 경도, 데이터 발송 시각 및 속도를 입력 데이터로 사용하여, 비정상 선박 궤적을 식별하는 딥 클러스터링 모형 DCECL을 제안하였다.

제안된 모형의 성능을 측정하기 위하여, 정상 궤적 데이터의 일부 데이터를 회전하여 회전 비정상 궤적 데이터 셋을 생성하였으며, 아울러 정상 궤적의 위치 데이터는 그대로 두고, 속도 데이터에 변화를 준 랜덤 속도 비정상 궤

적 데이터 셋을 만들었다.

제안된 모형과 오토인코더 기반의 세 종류 모형을 비정상 궤적을 이용하여 식별률을 비교한 결과, 랜덤 속도 비정상 궤적에 대해서는 제안된 모형이 비교한 세 모형보다 뛰어난 식별률을 보였다. 제안된 모형은 회전 비정상 궤적에 대하여 나머지 두 모형에 비하여 더 나은 식별률을 보였다.

본 논문은 임계값의 선택 기준이 임의로 설정 되었다는 한계를 갖는다. 본 논문에서 (평균+3시그마)로 정의한 복원 임계값과 클러스터 중심까지의 거리를 정렬하여 하위 2%로 정의한 클러스터링 임계값은, 제시한 모형을 실제로 배치하여 사용하는 경우 해안 경계 단위의 인원, 경계 영역의 크기 등을 고려하여 실정에 맞게 조정할 필요가 있다. 아울러 제시한 모형에서 사용한 K-means 알고리즘 외에도 Gaussian Mixture Model(GMM)이나 DBSCAN 등을 이용한 클러스터링의 결과와도 비교할 필요가 있다.

딥 클러스터링에서 사용하는 네트워크로는, 논문에서 사용한 오토인코더 외에도, 변분 오토인코더, 생성적 적대 신경망 등이 있으며, 이들 네트워크가 비정상 선박 궤적을 식별하는 문제에 어떠한 성능을 발휘하는지 연구할 필요가 있다.

## References

- [1] Atluri, G., Karoatne, A., and Kumar, V., Spatio-Temporal Data Mining: A Survey of Problems and Method, *ACM Computing Surveys*, 2018, Vol. 51, No. 4, Article 83.
- [2] Atmosukarto, I., Ghanem, B., and Ahuja, N., Trajectory-based Fisher Kernel Representation for Action Recognition in Videos, *21st International Conference on Pattern Recognition*, 2012, pp. 3333-3336.
- [3] Aytekin, C., Ni, X., Cricri, F., and Aksu, E., Clustering and Unsupervised Anomaly Detection with L2 Normalized Deep Auto-Encoder Representation, *arXiv:1802.00187[cs.LG]*, 2018.
- [4] Bian, J., Tian, D., Tang, Y., and Tao, D., A Survey on Trajectory Clustering Analysis, *arXiv:1802.06971[cs.CV]*, 2018.
- [5] Dhillon, I.S. and Modha, D.S., Concept Decompositions for Large Sparse Text Data using Clustering, *Machine Learning*, 2001, Vol. 42, No. 1, pp. 143-175.
- [6] Fu, P., Wang, H., Liu, K., Hu, X., and Zhang, H., Finding Abnormal Vessel Trajectories using Feature Learning, *IEEE Access*, 2017, Vol. 5, pp. 7898-7909.
- [7] Guo, X., Gao, L., Liu, X., and Yin, J., Improved Deep Embedded Clustering with Local Structure Preservation,

- International Joint Conference on Artificial Intelligence*, 2017, pp. 1753-1759.
- [8] Guo, X., Liu, X., Zhu, E., and Yin, J., Deep Clustering with Convolutional Autoencoders, *International Conference on Neural Information Processing*, 2017, pp. 373-382.
- [9] Li, S., Liang, M., and Liu, R.W., Vessel Trajectory Similarity Measure based on Deep Convolutional Autoencoder, *2020 5th IEEE International Conference on Big Data Analytics*, 2020, pp. 333-338.
- [10] Li, X., Zhao, K., Gong, G., Jensen, C.S., and Wei, W., Deep Representation Learning for Trajectory Similarity Computation, *IEEE 34th International Conference on Data Engineering*, 2018, pp. 617-628.
- [11] Liang, M., Liu, R.W., Li, S., Xiao, Z., Liu, X., and Lu, F., An Unsupervised Learning Method with Convolutional Auto-Encoder for Vessel Trajectory Similarity Computation, arXiv:2101.03169 [cs.LG], 2021.
- [12] Min, E., Guo, X., Liu, Q., Zhang, G., Cui, J., and Long, J., A Survey of Clustering with Deep Learning: From the Perspective of Network Architecture, *IEEE Access*, 2018, Vol. 6, pp. 39501-39514.
- [13] Naftel, A. and Khalid, S., Motion Trajectory Learning in the DFT-Coefficient Feature Space, *Fourth IEEE International Conference on Computer Vision Systems*, 2006, pp. 40-47.
- [14] Oh, J.-Y., Kim, H.-J., and Park, S.-K., Detection of Ship Movement Anomaly using AIS Data: A Study, *Journal of Navigation and Port Research*, 2018, Vol. 42, No. 4, pp. 277-282.
- [15] Oh, J.-Y., Kim, H.-J., and Park, S.-K., Development of a Decision Support System based on Autoencoder for Vessel Traffic Service, *KSIIE Transactions on Computing Practices*, 2018, Vol. 24, No. 12, pp. 642-648.
- [16] Olive, X., Basora, L., Viry, B., and Alligier, R., Deep Trajectory Clustering with Autoencoders, *Proceedings of the International Conference on Research in Air Transportation*, 2020, pp. 1-8.
- [17] Park, J. and Kim, S., Maritime Anomaly Detection Based on VAE-CUSUM Monitoring System, *Journal of the Korean Institute of Industrial Engineers*, 2020, Vol. 46, No. 4, pp. 432-442.
- [18] Rezaei, M., Yang, H., and Meinel, K., Deep Neural Network with l2-norm Unit for Brain Lesions Detection, arXiv:1708.05221[cs.CV], 2017.
- [19] Santhosh, K. K., Dogra, D. P., Roy, P. P., and Mitra, A., Video Trajectory Classification and Anomaly Detection using Hybrid CNN-VAE, ArXiv: 1812.07203[cs.CV], 2018.
- [20] Son, J.-H., Jang, J.-G., Choi, B., and Kim, K., Detection of Abnormal Vessel Trajectories with Convolutional Autoencoder, *Journal of Society of Korea Industrial and Systems Engineering*, 2020, Vol. 43, No. 4, pp. 190-197.
- [21] Taghizadeh, S., Elekes, A., Schaler, M., and Bohm, K., How Meaningful are Similarity in Deep Trajectory Representations?, *Information Systems*, 2021, Vol. 98, Article 101452.
- [22] Wilson, R.C., Hancock, E.R., Pekalska, E., and Duin, R.P.W., Spherical and Hyperbolic Embedding of Data, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014, Vol. 36, No. 11, pp. 2255-2269, 2014.
- [23] Xie, J., Girshick, R., and Farhadi, A., Unsupervised Deep Embedding for Clustering Analysis, *Proceedings of the 33rd International Conference on Machine Learning*, 2016, pp. 478-487.
- [24] Zhang, Z., Huang, K., and Tan, T., Comparison of Similarity Measures for Trajectory Clustering in Outdoor Surveillance Scene, *18th IEEE International Conference on Pattern Recognition*, 2006, pp. 1135-1138.

#### ORCID

Heon-Jei Park | <https://orcid.org/0000-0003-2512-4950>  
 Jun Woo Lee | <https://orcid.org/0000-0003-0248-8639>  
 Ji Hoon Kyung | <https://orcid.org/0000-0002-0359-5594>  
 Kyeongtaek Kim | <https://orcid.org/0000-0002-5636-2487>