Big-data-driven vessel destination prediction for smart port management

(Engineering Applications of Artificial Intelligence 2025, Chen et al)

MinJae Jeong

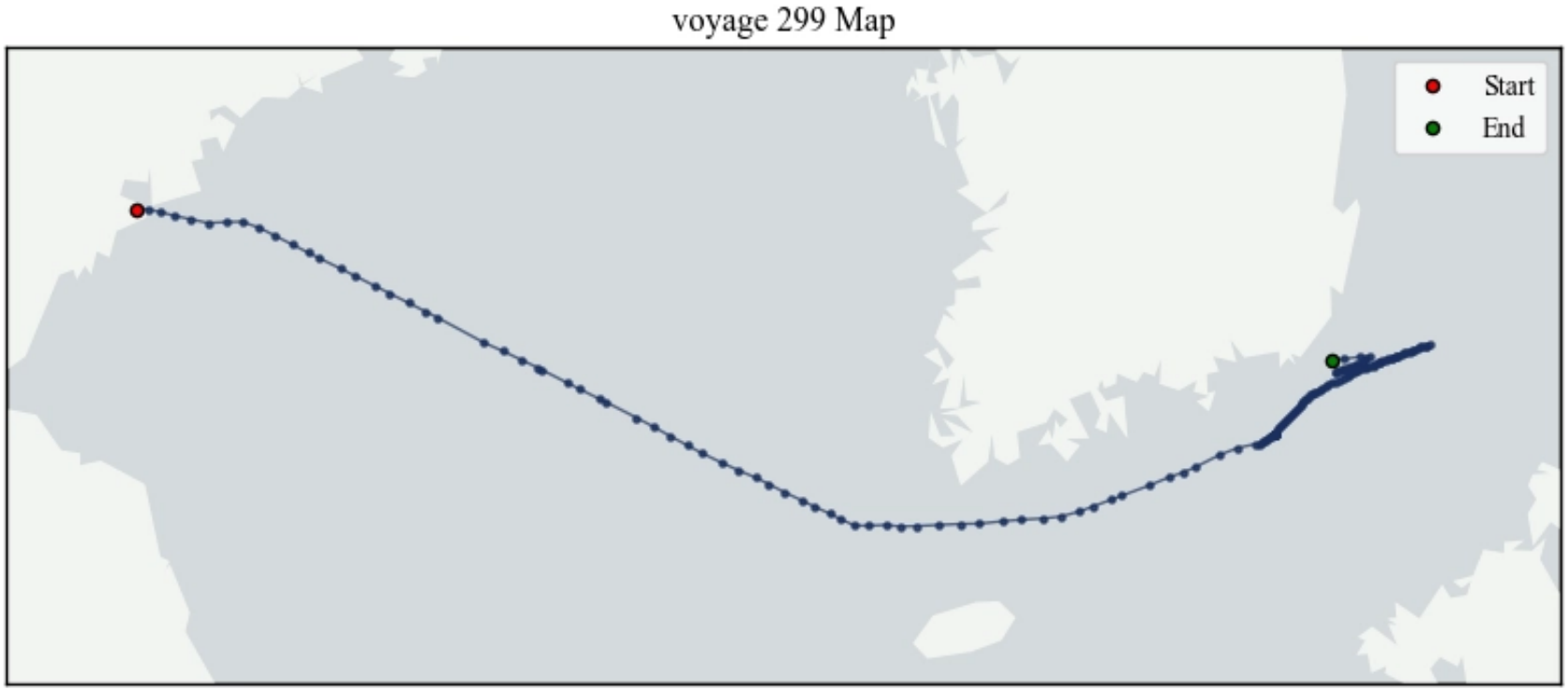
mj4788@pucan.ac.kr

Contents

Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Background

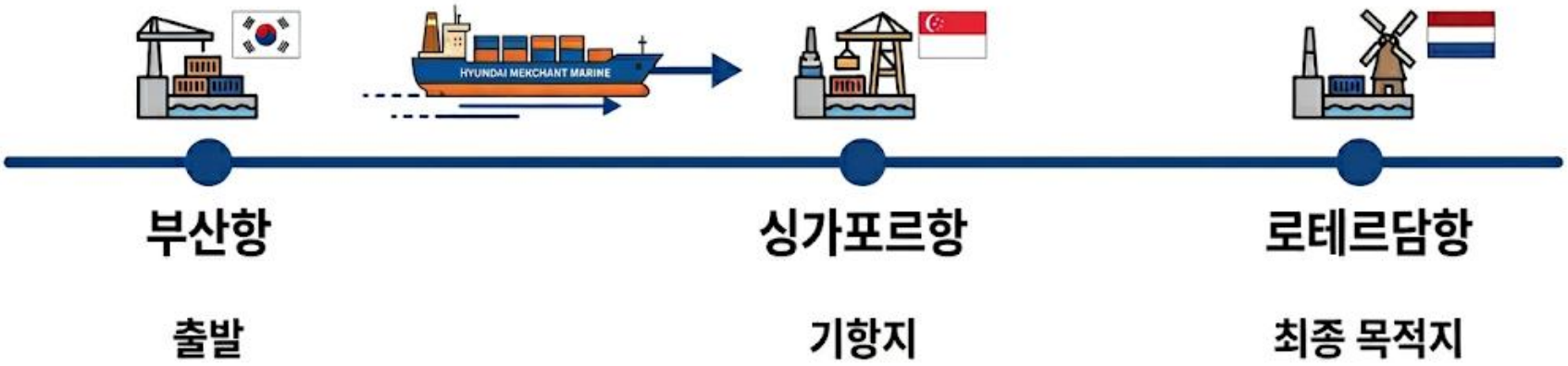
- **AIS Data**
  - AIS 데이터는 선박의 위치, 속도, 방향, 시간 정보를 지속적으로 제공하는 해상 운항 데이터이다.
  - 선박의 이동 경로와 운항 패턴을 파악할 수 있으며, 항만 도착 예측이나 해상 교통 분석에 활용된다.



Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Background

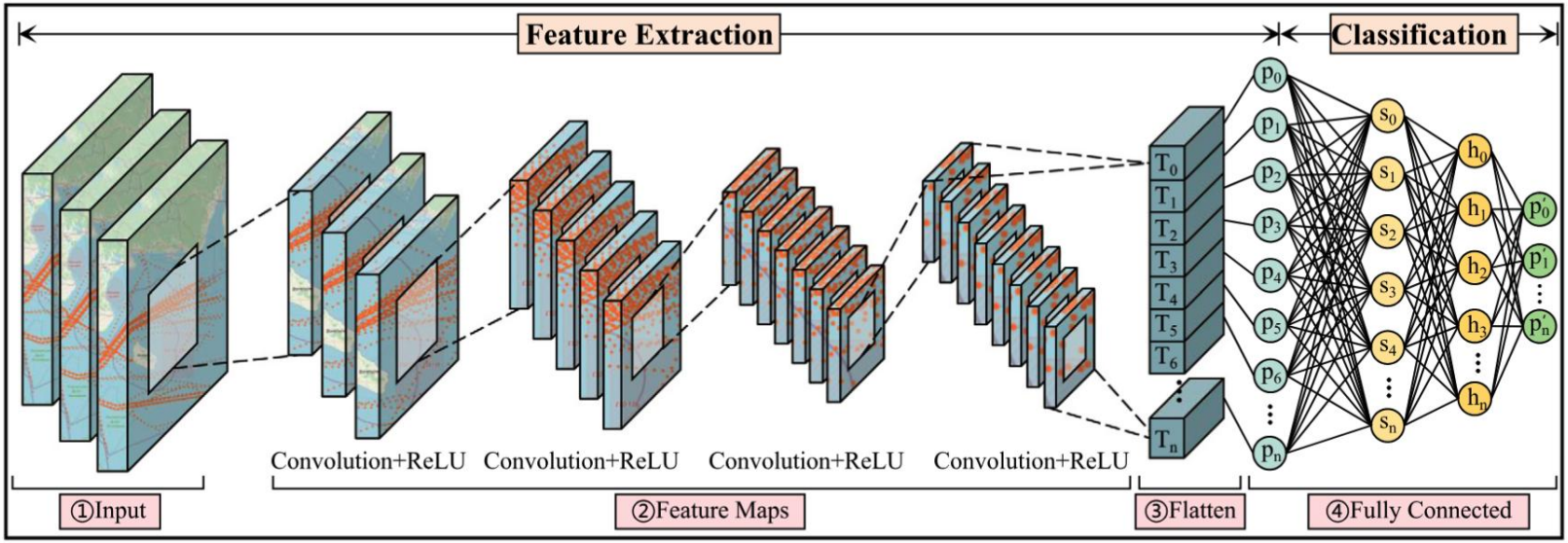
- **Vessel destination prediction**
  - AIS Data에 입력된 목적지 정보는 실제 도착 항만과 일치하지 않는 경우가 많고, 일부 연구에서는 정확도가 38% 수준까지 낮게 나타났다 (Yin et al., 2022).
  - 이러한 정보 불확실성은 선박 교통 관리와 항만 운영 최적화를 어렵게 만든다.
  - 따라서 선박의 실제 목적지를 정확히 예측하는 기술은 선석 배정, 혼잡 완화, 항만 자원 운영 효율화를 위해 중요하다.



Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Background

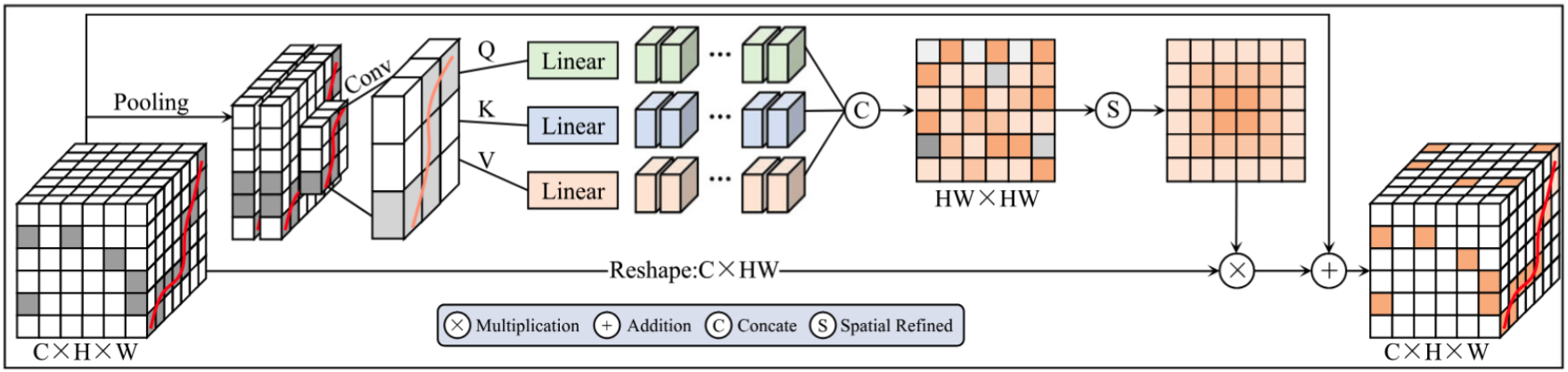
- **CNN (Convolutional Neural Networks)**
  - CNN은 이미지에서 중요한 패턴을 자동으로 추출해 분류하는 딥러닝 모델이다.
  - 합성곱 연산은 입력 이미지의 국소적인 특징을 찾아내고, 활성화 함수와 풀링은 비선형성을 부여하면서 핵심 정보만 더 잘 남기도록 한다.
  - 이후 추출된 특징은 fully connected layer를 거쳐 최종적으로 각 클래스에 대한 예측 결과로 변환된다.
  - 이 논문에서는 선박 AIS 궤적을 이미지로 변환한 뒤, CNN을 기반으로 항적과 항만 분포의 공간적 특징을 학습하여 목적지 항만을 분류한다.



Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Background

- **Attention**
  - Attention은 입력 정보 중에서 주어진 문제 해결에 더 중요한 부분에 높은 가중치를 부여하는 방식이다.
  - 이 과정은 각 위치의 중요도를 계산하고, 이를 정규화한 뒤, 중요한 특징이 더 크게 반영되도록 만든다.
  - 따라서 Attention은 전체 정보를 동일하게 다루지 않고, 예측에 핵심적인 부분에 선택적으로 집중하도록 한다.
  - 이 논문에서는 Attention을 활용해 선박 항적의 종점 구간과 항만 위치 특징에 더 큰 학습 가중치를 부여함으로써 목적지 항만 분류 성능을 높인다.



Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Related work

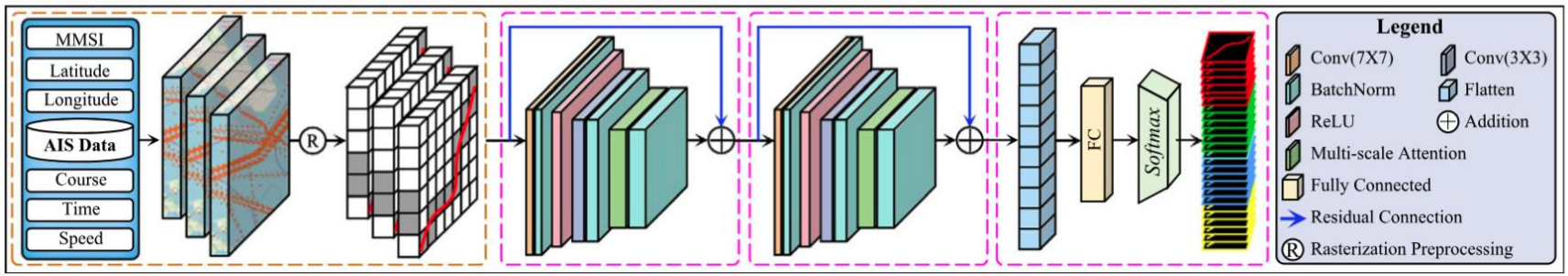
- **Clustering-based methods**
  - Clustering-based methods는 AIS 항적들 사이의 공간적 유사성을 분석하여 비슷한 경로를 그룹화함으로써 목적지 예측 문제를 단순화한다.
  - DBSCAN과 같은 군집화 기법이 비선형 항적 곡선을 효과적으로 묶어 선박의 이동 패턴과 목적지 특성을 반영하는 데 활용된다.
- **Learning-based methods**
  - Learning-based methods는 과거 항적 데이터로부터 선박 이동의 공간적·시간적 의존성을 학습하여 목적지를 예측한다.
  - decision tree, SVR, LSTM, GRU, TCN, GAN, graph-based model 등 다양한 학습 기반 방법이 복잡한 항적 패턴과 선박 간 상호작용을 효과적으로 모델링하는 데 활용된다.
- **Research gap**
  - 기존 연구들은 항적의 시간적 특성에 주로 집중하여 공간적 구조와 최종 목적지 항만 자체를 충분히 반영하지 못했으며, 대규모 AIS 데이터에서 효율적이면서도 정밀한 목적지 예측 모델은 여전히 부족하다.

Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Methodology

Overview

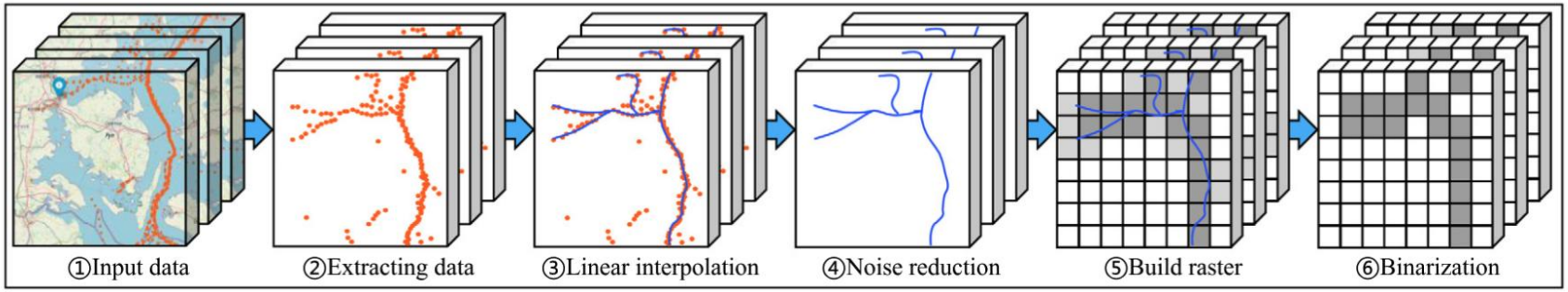
- 본 방법론은 AIS 기반 선박 항적 데이터를 전처리하여 목적지 항만 분류 문제로 변환하는 구조를 가진다.
- 먼저 선박의 시공간 항적 데이터를 동일한 위도·경도 범위의 평면 이미지로 변환하고, 필터링을 통해 항적 이미지를 정제한다.
- 이후 multi-scale residual convolutional network가 서로 다른 시야 범위에서 항적과 항만 분포의 특징을 추출한다.
- 마지막으로 multi-scale attention mechanism이 종점 항적과 항만 위치의 중요 특징에 더 큰 가중치를 부여하여 목적지 항만을 최종 분류한다.



Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Methodology

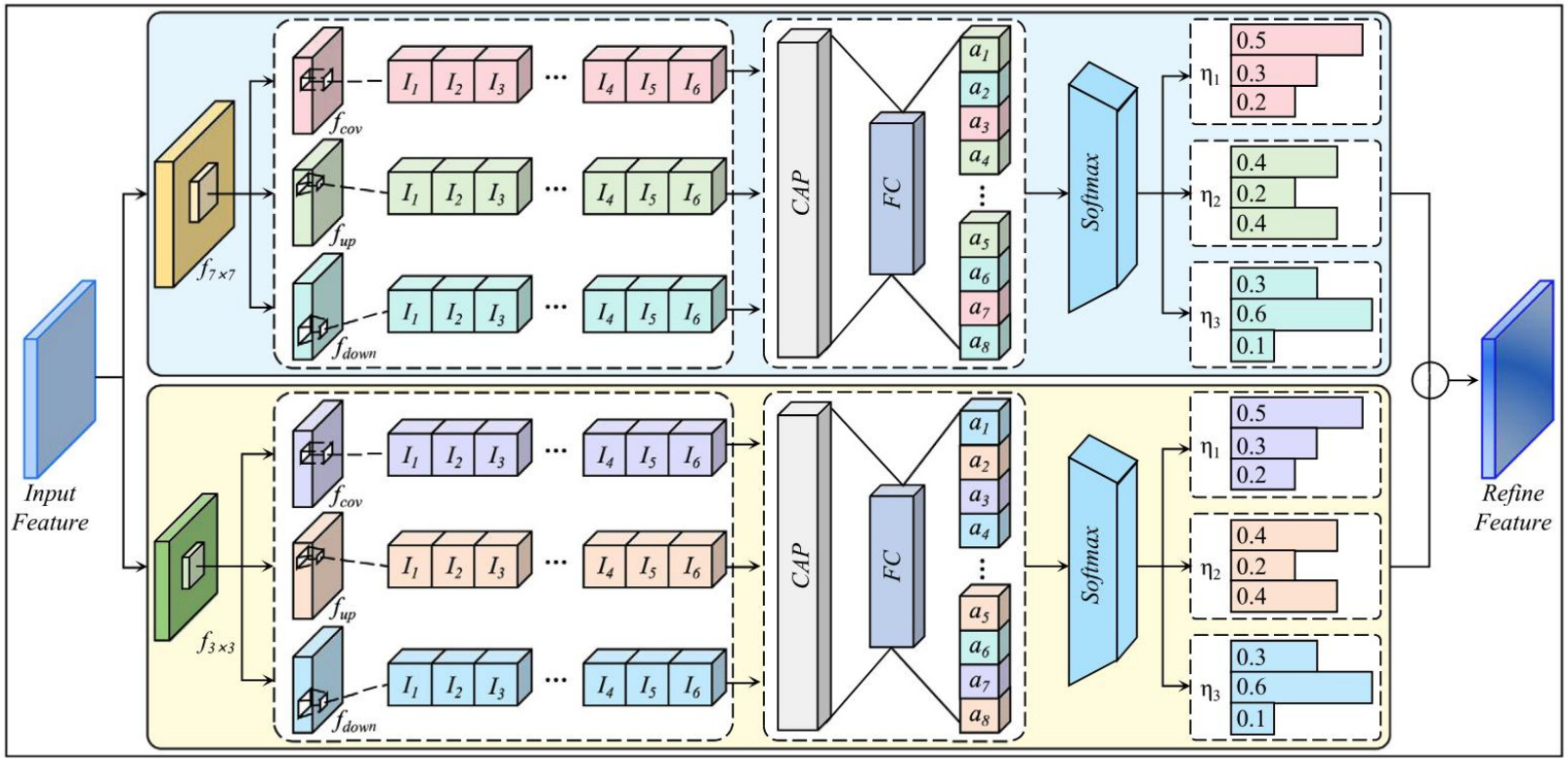
- **Trajectory image rasterization preprocessing**
  - AIS 항적 데이터를 이미지 형태로 바꾸어 CNN이 학습할 수 있도록 만드는 전처리 과정이다.
  - 먼저 선박의 이동 좌표를 같은 시간 간격에 맞춰 다시 보정하여, 끊기거나 불규칙한 경로를 더 자연스럽게 연결한다.
  - 이후 가우시안 필터를 적용해 흔들림과 잡음을 줄인 뒤, 항적을 격자 위에 배치해 이미지처럼 표현한다.
  - 마지막으로 이진화를 통해 항적만 남는 단순한 흑백 이미지로 변환하여 핵심 경로 특징이 더 잘 드러나도록 한다.



Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Methodology

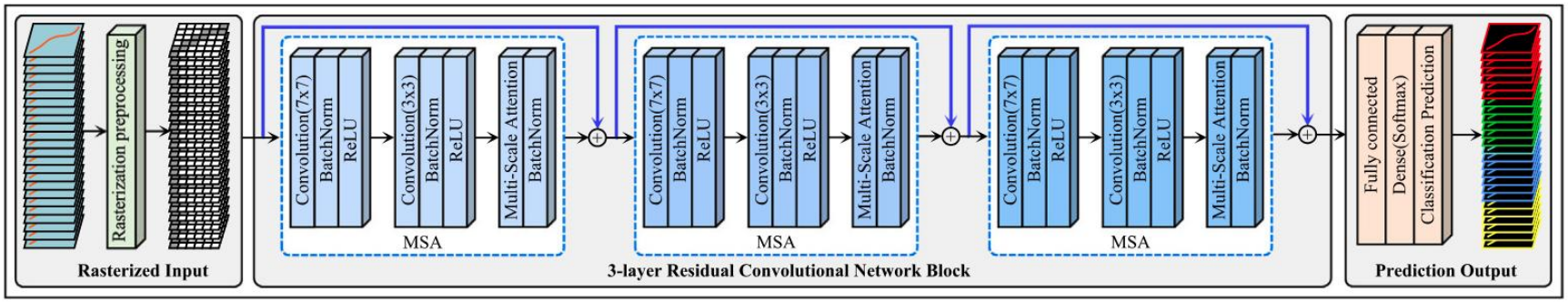
- **Multi-scale attention mechanism**
  - 서로 다른 스케일의 특징을 함께 반영하면서, 목적지 예측에 중요한 종점 항적과 항만 위치 특징에 더 큰 가중치를 부여하는 구조이다.
  - 이를 통해 선박의 마지막 접근 방향을 더 효과적으로 학습하여 목적지 항만 분류 성능을 높인다.



Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Methodology

- **Multi-scale residual convolutional network**
  - 이 네트워크는 전처리를 거친 항적 이미지를 입력으로 받아 목적지 항만의 분류 확률을 예측하는 구조이다.
  - 모델은 3개의 multi-scale residual unit으로 구성되며, 각 단위는 서로 다른 크기의 convolution과 attention layer를 결합해 종점 항적과 항만 특징을 학습한다.
  - 또한 각 블록은 residual connection을 통해 이전 특징을 다음 층으로 전달함으로써 학습 안정성을 높이고 더 풍부한 다중 스케일 정보를 유지한다.
  - 최종적으로 추출된 특징은 fully connected layer와 softmax를 거쳐 각 목적지 항만 클래스의 확률로 출력된다.



Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Methodology Code

```python
class MultiScaleAttention(nn.Module): # Multi-scale attention mechanism
    def __init__(self, channels):
        super().__init__()
        self.conv_main = nn.Conv2d(channels, channels, kernel_size=7, padding=3) # 원본 스케일 특징을 추출하는 7×7 convolution
        self.conv_down = nn.Conv2d(channels, channels, kernel_size=7, padding=3) # 축소된 입력에서 전역적 특징을 추출하는 7×7 convolution
        self.conv_up = nn.Conv2d(channels, channels, kernel_size=7, padding=3) # 확대된 입력에서 세부적 특징을 추출하는 7×7 convolution
        self.fc = nn.Linear(channels, 1) # 각 스케일 특징맵의 중요도 score를 계산하는 fully connected layer

    def forward(self, x):
        f1 = self.conv_main(x) # 원본 스케일에서 첫 번째 특징맵 생성

        x_down = F.avg_pool2d(x, kernel_size=2, stride=2) # 평균 풀링으로 입력을 절반으로 축소하여 더 넓은 시야의 특징을 반영
        f2 = self.conv_down(x_down) # 축소된 스케일에서 두 번째 특징맵 생성
        f2 = F.interpolate(f2, size=x.shape[-2:], mode="bilinear", align_corners=False) # 다른 스케일과 결합할 수 있도록 원래 크기로 복원

        x_up = F.interpolate(x, scale_factor=2, mode="bilinear", align_corners=False) # 입력을 2배 확대하여 더 세밀한 공간 패턴을 강조
        f3 = self.conv_up(x_up) # 확대된 스케일에서 세 번째 특징맵 생성
        f3 = F.interpolate(f3, size=x.shape[-2:], mode="bilinear", align_corners=False) # 최종 융합을 위해 원래 크기로 다시 맞춤

        a1 = self.fc(F.adaptive_avg_pool2d(f1, 1).flatten(1)) # 원본 스케일 특징의 전역 요약값으로 중요도 score 계산
        a2 = self.fc(F.adaptive_avg_pool2d(f2, 1).flatten(1)) # 축소 스케일 특징의 전역 요약값으로 중요도 score 계산
        a3 = self.fc(F.adaptive_avg_pool2d(f3, 1).flatten(1)) # 확대 스케일 특징의 전역 요약값으로 중요도 score 계산

        alpha = torch.softmax(torch.cat([a1, a2, a3], dim=1), dim=1) # 세 스케일의 중요도를 softmax로 정규화하여 attention weight 생성

        out = (
            alpha[:, 0].view(-1, 1, 1, 1) * f1 +
            alpha[:, 1].view(-1, 1, 1, 1) * f2 +
            alpha[:, 2].view(-1, 1, 1, 1) * f3
        )
        return out # 정규화된 가중치로 세 스케일 특징맵을 가중합하여 최종 특징맵 생성
```

Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Methodology Code

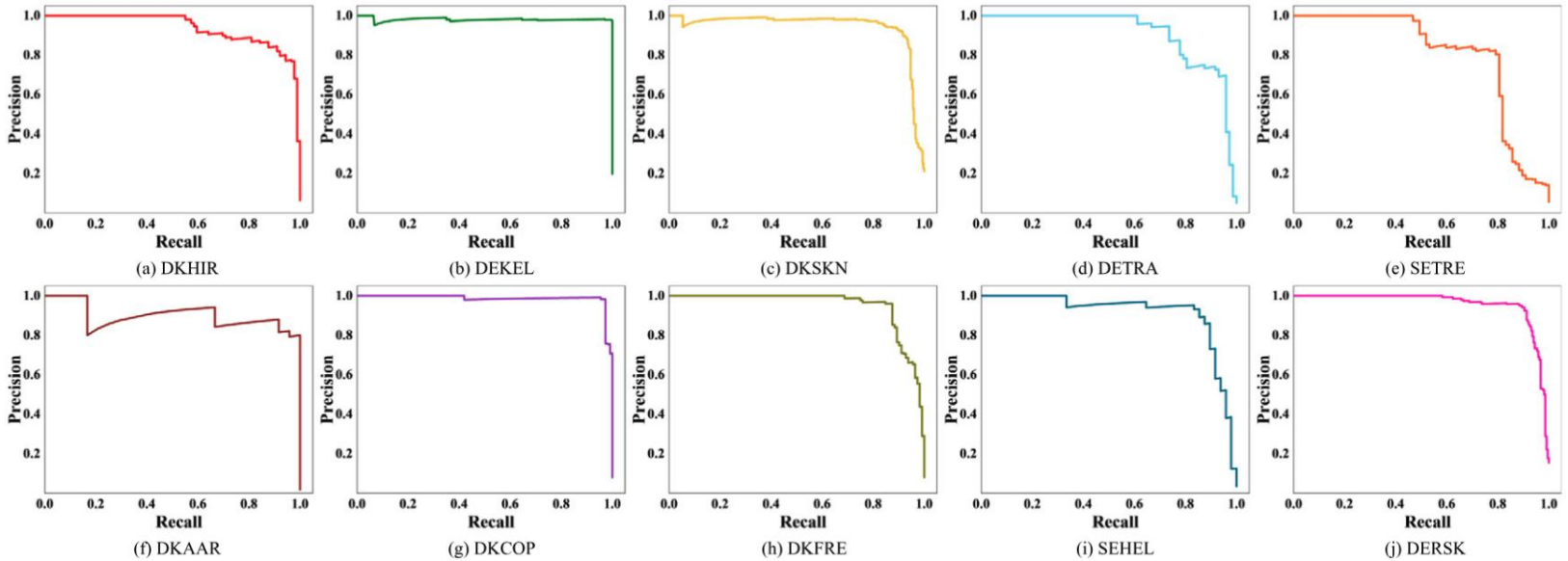
```python
class MultiScaleResidualUnit(nn.Module): # Multi-scale residual convolutional
    def __init__(self, channels):
        super().__init__()
        self.conv7 = nn.Sequential( # 넓은 receptive field로 전역적 패턴을 추출하는 7×7 convolution branch
            nn.Conv2d(channels, channels, kernel_size=7, padding=3), # 7×7 convolution으로 큰 범위의 공간 특징 추출
            nn.BatchNorm2d(channels),
            nn.ReLU(inplace=True)
        )
        self.conv3 = nn.Sequential( # 작은 receptive field로 세부 패턴을 추출하는 3×3 convolution branch
            nn.Conv2d(channels, channels, kernel_size=3, padding=1), # 3×3 convolution으로 국소적 공간 특징 추출
            nn.BatchNorm2d(channels),
            nn.ReLU(inplace=True)
        )
        self.attn = MultiScaleAttention(channels) # Multi-scale attention mechanism
        self.fuse = nn.Sequential( # 여러 branch 출력을 하나의 특징맵으로 통합하는 결합층
            nn.Conv2d(channels * 3, channels, kernel_size=1), # concat된 채널을 1×1 convolution으로 압축 및 융합
            nn.BatchNorm2d(channels),
            nn.ReLU(inplace=True)
        )

    def forward(self, x):
        f7 = self.conv7(x) # 7×7 branch를 통해 전역적 특징맵 생성
        f3 = self.conv3(x) # 3×3 branch를 통해 국소적 특징맵 생성
        fa = self.attn(x) # 입력 x를 바탕으로 multi-scale attention feature 생성
        y = self.fuse(torch.cat([f7, f3, fa], dim=1)) # 세 feature를 채널 방향으로 결합한 뒤 하나의 feature로 통합
        return x + y # residual connection으로 현재 블록이 만든 새로운 특징 y에 원래 입력 x를 더함
```

Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Experiment

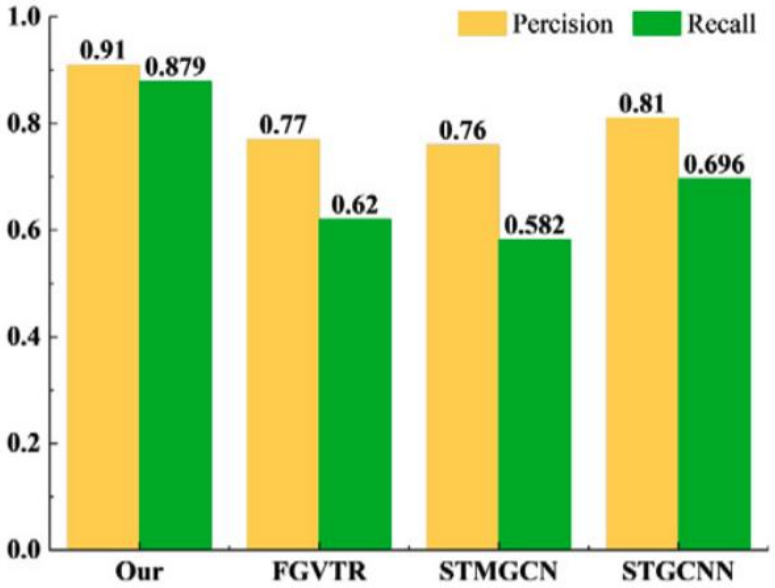
- **Evaluation of classification prediction performance**
  - PR curve를 보면 대부분의 항만에서 곡선이 높은 구간에 유지되어, 제안 모델이 전반적으로 안정적인 목적지 분류 성능을 보인다.
  - 특히 DEKEL, DKSKN, DKCOP는 우수한 성능을 보인 반면, DETRA, SEHEL, DERSK는 상대적으로 성능 저하가 나타나 항만별 예측 난이도 차이가 존재함을 확인할 수 있다.

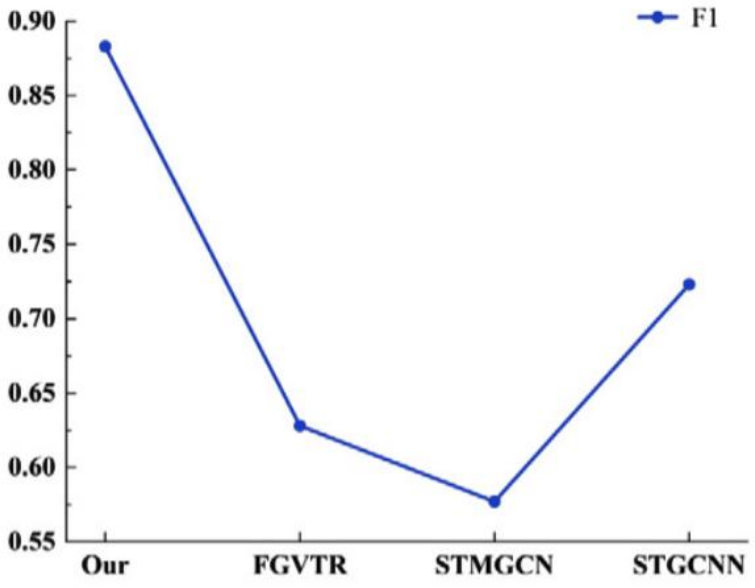


Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Experiment

- **Evaluation of classification prediction performance**
  - 비교 실험 결과, 제안 모델은 Precision 0.91, Recall 0.879, F1 0.88 수준을 기록하며 FGVTR, STMGCN, STGCNN보다 가장 우수한 분류 성능을 보인다.
  - 즉 제안 방법은 종점 항적과 항만 위치의 핵심 특징을 효과적으로 반영함으로써 기존 비교 모델들보다 목적지 항만 분류에 더 적합한 성능을 나타낸다.

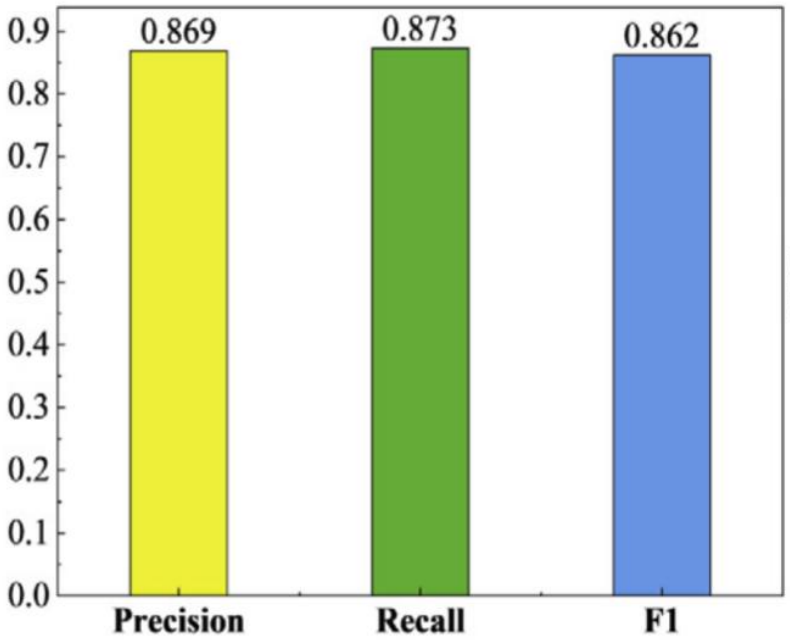


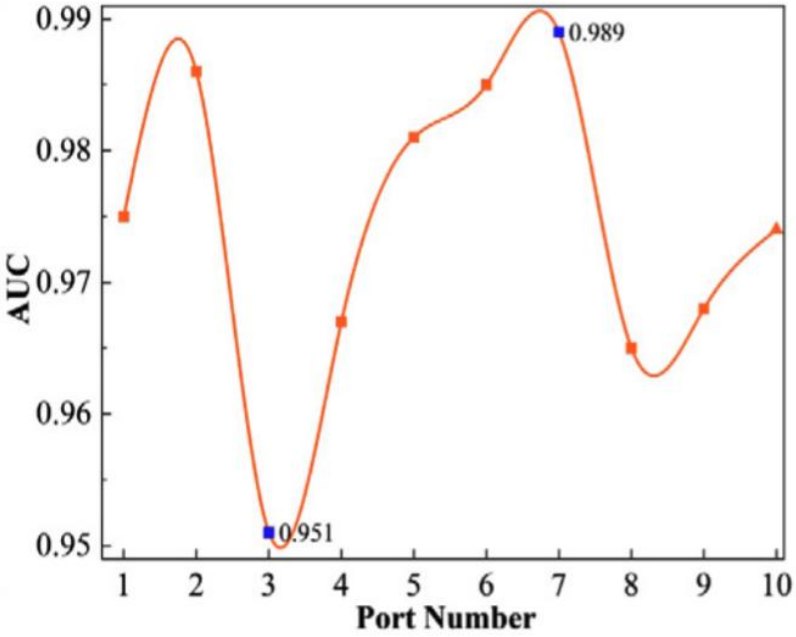


Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Experiment

- **Evaluation of classification prediction performance**
  - Cross-database experiment 결과, 제안 모델은 American dataset에서도 Precision 0.869, Recall 0.873, F1 0.862를 기록하여 Danish dataset에서 학습한 모델이 다른 해역에서도 일정 수준의 적용 가능성을 보임을 확인할 수 있다.
  - 또한 항만별 AUC도 대부분 0.97 이상을 유지하여, 제안 방법이 특정 데이터셋에만 최적화된 것이 아니라 다른 환경에서도 일정 수준의 일반화 가능성을 가짐을 시사한다.

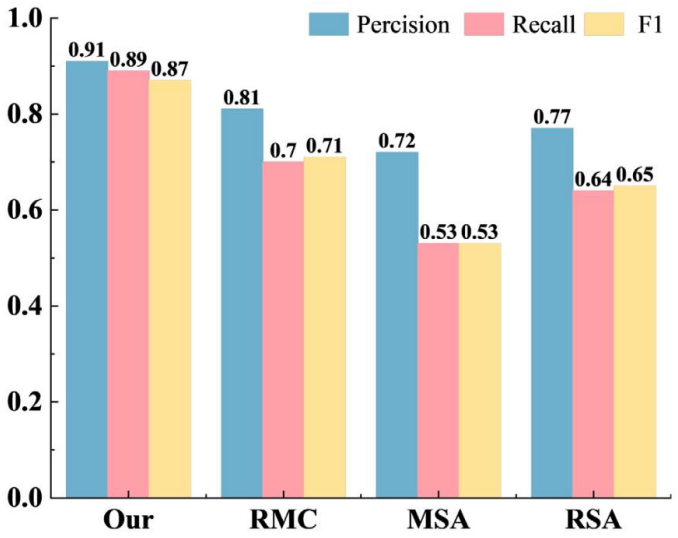


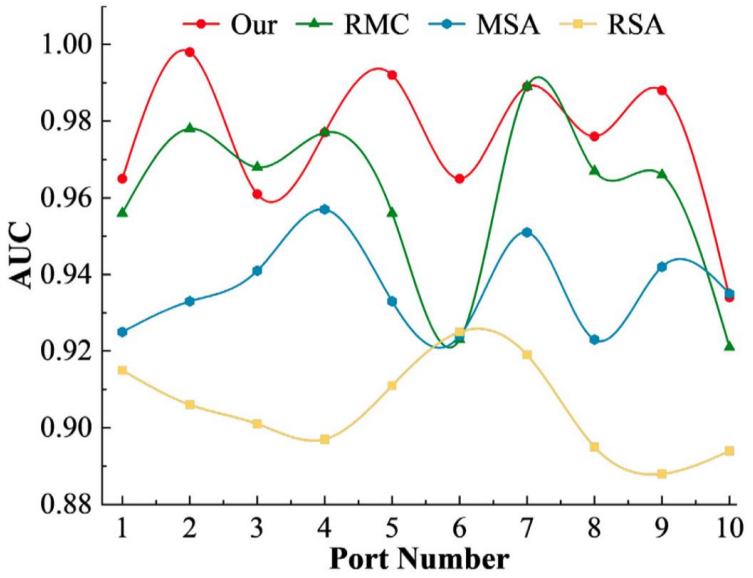


Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Experiment

- **Evaluation of classification prediction performance**
  - Ablation study 결과, 제안 모델은 Precision 0.91, Recall 0.89, F1 0.87로 RMC, MSA, RSA보다 모든 지표에서 가장 우수한 성능을 보인다.
  - 이는 rasterization preprocessing, multi-scale attention, residual structure가 각각 독립적으로도 의미가 있지만, 세 요소가 함께 결합될 때 목적지 항만 분류 성능이 가장 크게 향상됨을 보여준다.
  - 또한 항만별 AUC 비교에서도 제안 모델이 대부분의 Port Number에서 가장 높은 값을 유지하여, 전체 평균 성능뿐 아니라 항만별 예측 안정성 측면에서도 우수함을 확인할 수 있다.

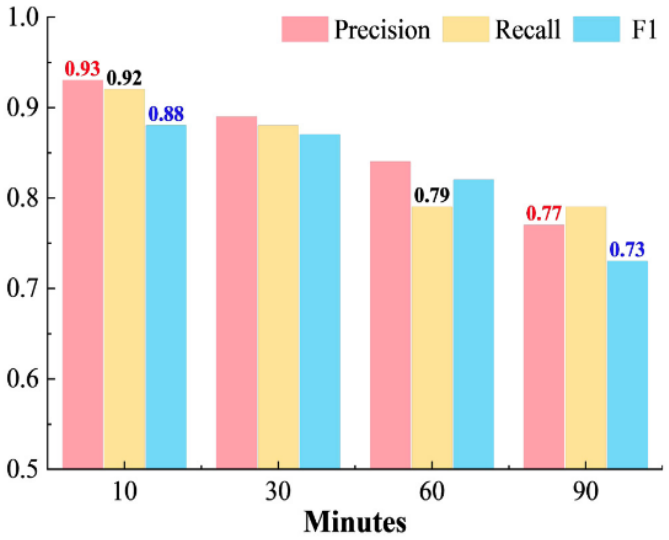


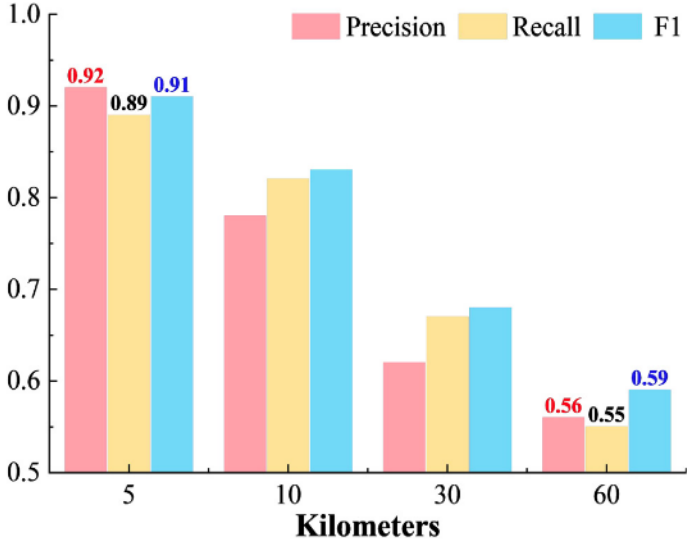


Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Experiment

- **Evaluation of classification prediction performance**
  - 항만에 가까운 시점과 거리에서의 항적일수록 분류 성능이 높게 나타나며, 도착 10분 전과 항만 5 km 부근에서 가장 높은 Precision·Recall·F1 값을 보인다.
  - 반대로 도착 시간이 더 많이 남아 있거나 항만에서 멀어질수록 성능이 점차 감소하며, 특히 60 km 거리에서는 분류 정확도가 매우 낮아진다.
  - 이는 항만에 가까워질수록 종점 항적과 항만 사이의 연관 특징이 뚜렷해 지지만, 먼 거리에서는 아직 공용 항로를 따라 이동하는 구간이 많아 목적지별 차이가 충분히 드러나지 않기 때문이다.

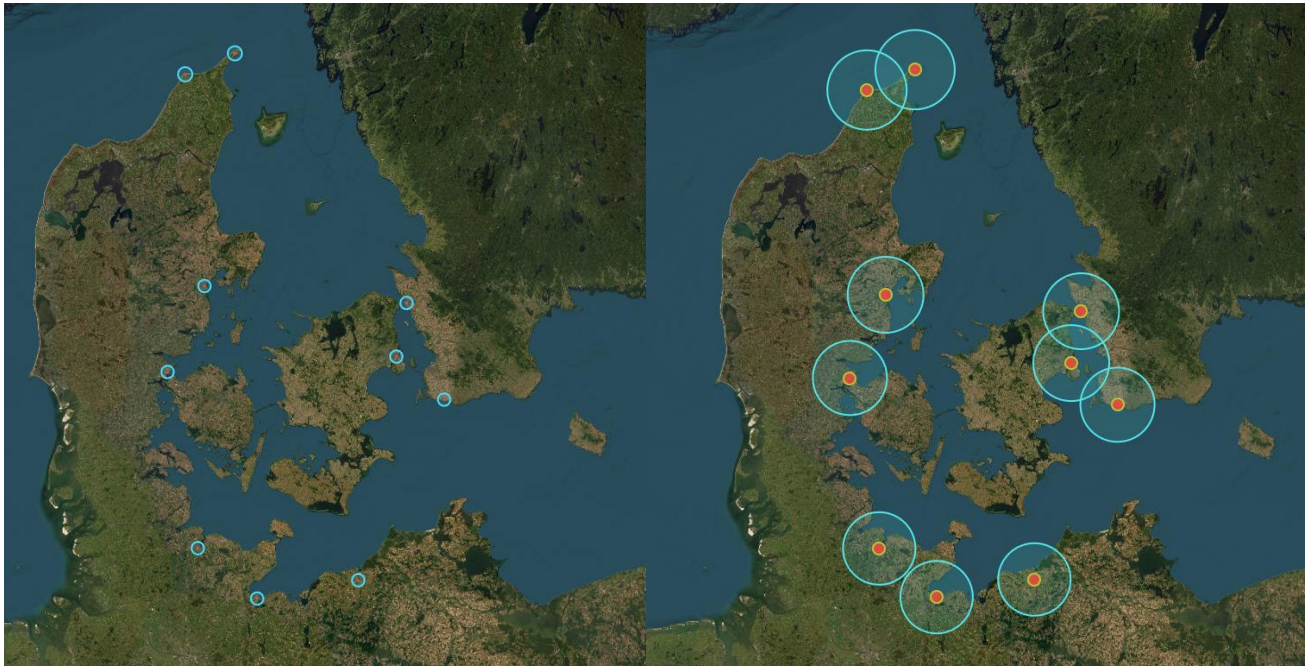




Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Limitation

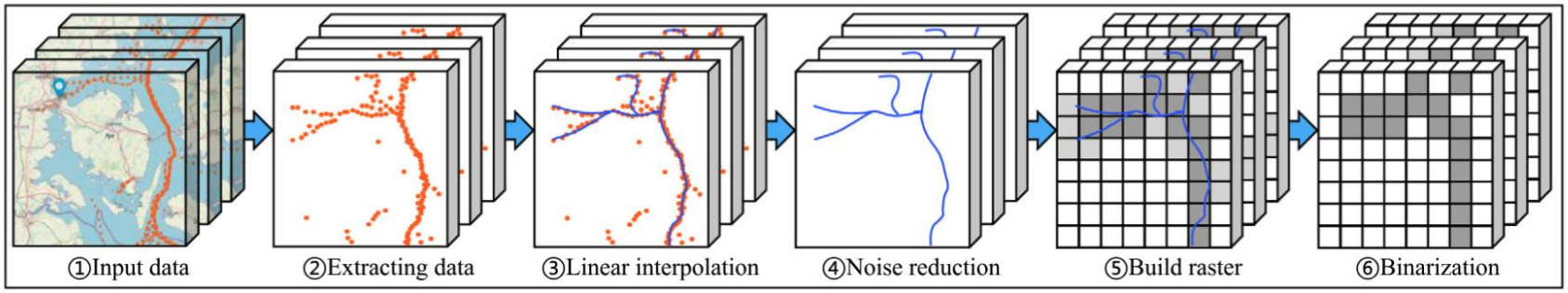
- 관련 연구에서 trajectory similarity 기반의 clustering method가 주요 접근으로 소개되었음에도 실제 비교 실험은 LSTM, GRU, STMGCN, FGVTR, STGCNN 등 딥러닝 학습 기반 모델에 집중되어 있어, 보다 보편적인 클러스터링 기반 방법과의 직접적인 성능 비교가 부족하다.
- 또한 논문 자체 결과에서도 항만 도착 시점이 멀어지거나 항만과의 거리가 커질수록 분류 성능이 뚜렷하게 감소하며, 특히 60 km 거리에서는 정확도가 매우 낮아져 장거리·조기 예측 상황에서의 실용성에는 한계가 있음을 보여준다.



Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Limitation

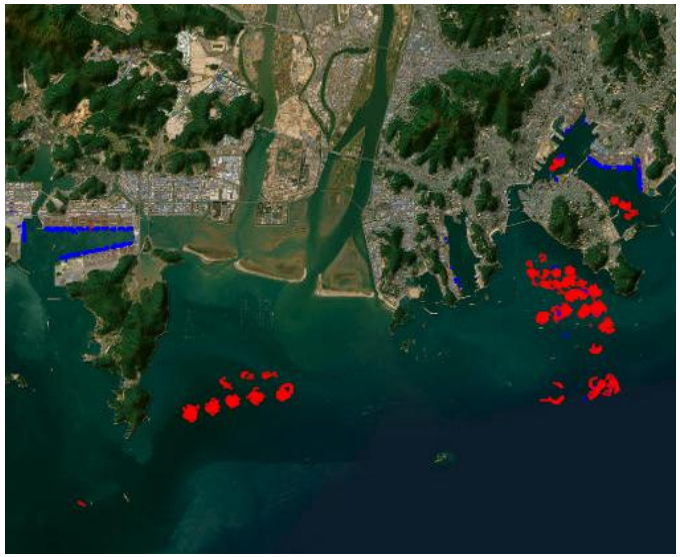
- 전처리 예시 그림은 단일 항적이라기보다 여러 갈래가 섞인 형태로 보여, 실제 입력 샘플이 무엇인지 직관적으로 이해하기 어렵다.
- 또한 논문은 입력을 흑백·이진 항적 이미지로 설명하면서도 일부 수식에서는 채널 수를 3으로 표기하여, 입력 표현 방식이 일관되지 않다.
- Ablation study 역시 RMC, MSA, RSA의 개념만 간단히 제시할 뿐, 실제 입력 구성과 세부 구현 과정이 충분히 설명되지 않아 재현성이 떨어진다.
- 일반화 검증도 Danish dataset 학습과 American dataset 검증의 단일 설정에 머물러 있어, 다양한 해역에 대한 확장 가능성을 충분히 입증했다고 보기는 어렵다.

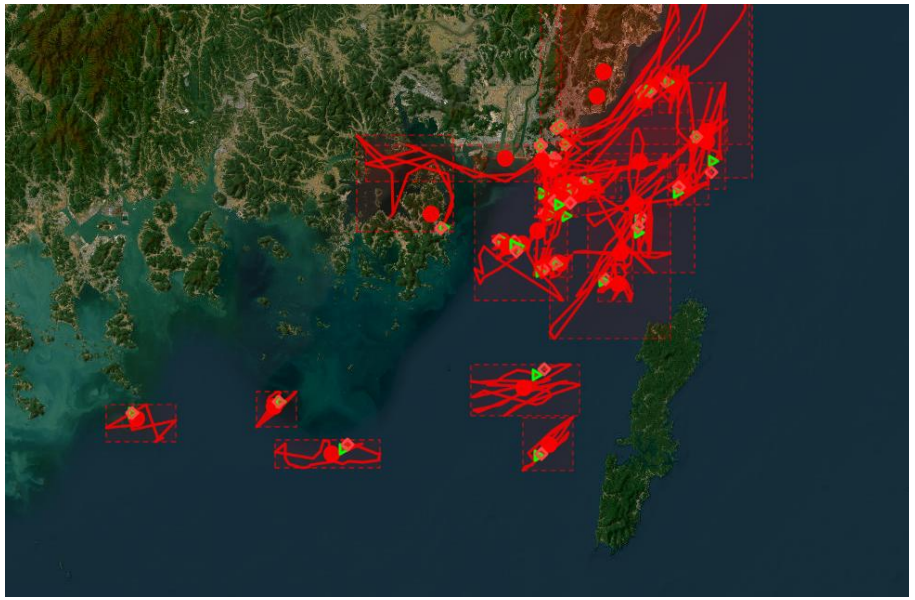


Prof. Hyerim Bae (hrbae@pusan.ac.kr)

Future work

- Future work로는, 논문이 종점 근처 패턴 학습에 강점을 보였다는 점을 바탕으로 장거리 구간에서는 similarity·clustering 기반 방법으로 목적지 항만 후보를 먼저 좁히고, 항만 근접 이후에는 raster image 기반 CNN으로 묘박, 배회, 접근과 같은 선박 상태를 식별하는 2-stage 구조를 고려할 수 있다.
- 이와 같은 구조는 CNN을 전체 목적지 예측 모델로 확장하기보다, 항만 근접 구간에서 최종 분류를 보조하는 현실적인 활용 방향이 될 수 있다.
- 즉 유사도 기반 방법의 후보 축소 능력과 CNN의 정밀한 근접 구간 패턴 분류 능력을 함께 활용하는 방향으로 발전시킬 수 있다.





Source: 신입생스터디_JM

Prof. Hyerim Bae (hrbae@pusan.ac.kr)