**TDOA 확률화 이미지를 활용한 이동하는 물체 위치 추적 알고리즘**

**TDOA Probabilisticization Image based Moving-target Tracking with Convolutional-LSTM**

**Abstract**

하드웨어 기술이 발전하면서 UWB(Ultra WideBand)를 이용해 무인 가게, 스마트 키 등 정밀 측위에 사용되고 있다. 그 중, TDOA(신호 도착 차이)는 병렬적 처리와 작은 배터리 소모로 인해 범용적으로 사용되는 측위 방식이다. 하지만 기존 방식은 장애물이 많은 환경일 경우 위치 파악 능력이 현저히 떨어진다. 이를 보완하기 위해 연속 신호에서 딥러닝을 활용해 노이즈를 제거한 TDOA 값을 찾아내는 연구가 진행되었다. 본 논문에서는 이와 다르게 노이즈를 포함하고도 이동하는 물체의 정확한 위치를 찾을 수 있는 방법에 대해 실험을 진행한다. 본 실험은 다양한 환경에서의 일반성과 이미지 표현의 정밀성을 향상시키기 위해 이진적 표현이 아닌 확률적 표현을 이용해 TDOA 이미지를 변환한다. 그런 다음 연속적인 시계열 TDOA 이미지를 입력값으로 하는 ‘ConvLSTM’ 모델을 훈련시켜 실시간으로 물체의 좌표를 추정한다. 시뮬레이션 결과, 장애물의 유무와 상관없이 'ConvLSTM '모델은 타겟의 이동에 대한 연속성을 파악하여 가장 좋은 결과를 보인다. 또한, 이미지로 변환하는 과정을 통해 신호 오차의 영향력이 감소하여 물체의 운동성 또한 파악할 수 있는 상태로 변환된다. 이로 인해 다양한 환경, 특히 복잡한 환경에서 보편적으로 사용될 실내 측위 기술이라 기대한다.

**1. Introduction**

\* 위치 파악 기술 대중성

인공위성을 통해 물체의 위치를 파악하는 위치추적 기술은 현재 우리 삶에서 많이 사용되고 있다. 네비게이션이나 스마트폰 등 일상은 물론 물류 창고나 선박, 비행기, 자동차 등 실시간 위치 추적이 필요한 모빌리티 전반에 위치 추적 기술이 활용되고 있으며 거의 모든 전자기기에 해당 기술이 탑재되어 있다. 이때 널리 사용되는 기술은 글로벌 네비게이션 위성 시스템(Global Navigation Satellite System(GNSS) / 대한민국에선 GPS로 통칭되며 GPS, GLONASS, Beidou, Galileo 등이 해당됨)이다.

\* 좁고 디테일한 위치 파악 기술 필요성

하지만 GNSS 기술엔 큰 한계가 있었는데, 실내에서의 위치 확인이 쉽지 않다는 것이다[locata,,]. 이러한 어려움을 해결하고자 Wi-Fi와 Bluetooth, UWB 등 다양한 기술들을 사용해 실내에서의 정확한 위치 파악을 하는 연구가 진행되고 있다.

\*UWB 기술 종류 소개

UWB는 짧은 시간의 낮은 출력의 펄스 신호를 사용하여 500MHz 이상의 넓은 주파수 대역으로 데이터를 송수신하는 100m 이내의 무선통신 기술이다. 해당 기술은 1970년대 군사 목적으로 개발되었으며 2000년대 이후 802.15.4a라는 이름으로 IEEE에서 발표되었다[1]. UWB는 수 cm 이내의 측위 정확성, Multipath 등의 간섭에 대한 저항성, 측정값에 대한 강건함, PHY layer와 난수를 활용한 보안성 및 실시간성의 장점을 지니고 있다. 하지만 그 당시 경쟁 기술이었던 Wi-Fi에 비해 미흡한 ecosystem과 하드웨어 기술 부족으로 인해 주목받지 못했다[2]. 시간이 흘러 칩 기술의 발달로 앞선 장점 때문에 재주목을 받았으며 현재 다양한 분야에서 활용되고 무선통신기술이다[3-5].

\*TDOA 기술의 장점 및 단점

UWB를 활용해 무선기기 간의 거리를 측정하는 방식은 크게 time of arrival (ToA)과 two way ranging(TWR)이 있다. 그 중, ToA는 송신기에서 빛을 쏘아 수신기에서 반사되어 돌아오는 시간을 측정한 뒤, 빛의 속도를 곱하여 거리를 구하는 방식이다. 하지만 ToA 방식은 태그(송신기)의 주소 바인딩이 필요해 통신할 수 있는 기기의 개수가 한정되어 있으며, 장치끼리 여러 번의 통신을 하기에 배터리 소모가 크다[6]. 이러한 문제를 보완하기 위해 서로 다른 앵커(수신기)에서 측정된 신호 시간의 차이를 이용하는 TDOA(Time Difference of Arrival) 방식을 주로 사용한다. TDOA를 이용한 위치 추적 방법을 사용시, 태그는 앵커와 개별적으로 통신하지 않고 Blink 메시지만 전송하므로 배터리 소모가 적다. 또한, 주소 바인딩이 없고 한 번의 재생 빈도로 수 많은 태그에 신호를 전송하기에 동시에 많은 태그와 통신할 수 있다. 그러나 이러한 방식은 복잡한 비선형 방정식 해를 찾는데 많은 컴퓨터 자원이 소모되는 단점을 지닌다.

\* 고전적 TDOA를 활용한 위치 추적 방법 제시 및 해당 기술의 단점

이러한 비선형 방정식의 계산 복잡성을 해결하고자 Chan, Taylor, Kalman Filter 등을 활용한 최적화 알고리즘이 제안되었다.[7-9]. 해당 방법들을 통해 연산 속도는 빨라졌지만 여전히 수식에 기반하여 위치를 추정하기에, 입력 값의 noise에 대해 민감하게 반응한다. 그 결과, 해당 방식은 장애물이 많은 환경에서 측정한 TDOA일 경우 측정 noise 때문에 물체의 위치를 전혀 다른 위치로 추론하는 경우가 빈번히 발생한다.

\* 머신러닝 및 딥러닝 을 활용한 위치 추적 방법

이러한 오류에 대해 강건한 위치 추정 모델을 만들기 위해 머신 러닝 기법을 활용한 방법이 사용되고 있다[10-12]. 특히 최근 들어 딥러닝을 활용한 방법이 각광을 받고 있다[13-16]. 이러한 방식들은 AI를 활용해 Additive White Gaussian Noise(AWGN, 가산성 백색 가우시안 잡음)가 제거된 TDOA를 측정하는 것을 목표로 하여 물체의 정확한 위치를 추론하고자 했다.

\* 새로운 방법 제안

선행 연구들은 정확한 TDOA를 얻기 위해 노력했지만, 이로 인해 여러 전처리 파이프라인을 거쳐 TDOA가 보정되는 복잡한 프로세스를 가진다. 또한, 환경이 복잡해질수록 신호 변형이 기하급수적으로 늘어나기에, 무작위성을 모두 고려해서 정확한 TDOA를 구할 수 있는 알고리즘 구현은 매우 어렵다. 마지막으로 특정 환경과 상황에 대해서만 검증하였기에 다양한 변수들이 존재하는 실제 환경과는 괴리감이 있다. 그 결과 장애물이 많이 발생하는 복잡한 공간에 실제로 도입하기엔 여전히 문제가 있다. 그렇기에 본 논문에서는 이러한 문제들을 해결하기 위해 노이즈가 많이 발생하는 공간에서 추출한 TDOA 값들을 보정하지 않고 그대로 활용하여 물체의 정확한 좌표를 추정하는 end-to-end모델을 제안한다. 특히 이번 연구는 물체가 특정 공간 안에서 운동을 하고 있다는 사실에 주목하여 장애물이 많은 좁은 공간에서 정확하게 물체의 위치를 파악할 방법을 찾고자 하였다.

**2. Related Work**

[blank line]

**1. Indoor Tracking**

\*TDOA 계산 과정 소개

TDOA 측위 시스템의 순서는 다음과 같다[16]. 1) UWB 태그가 메시지를 송신 2) 주변의 UWB 앵커가 이 메시지를 수신 3) 중앙 컴퓨팅 유닛(CCU)은 전체 UWB 앵커들의 데이터를 수집하여 각 앵커들 간 신호 도달시간 차이(TDOA)를 계산 4) TDOA를 이용해 쌍곡선 방정식을 만듦 5) 방정식들의 해를 계산하여 UWB 태그의 위치를 추정. 즉, TDOA는 앵커에 신호가 도착하는 시간의 차이이며, 물체의 위치는 초점이 앵커이며 거리 차는 TDOA인 쌍곡선 방정식을 통해 구한다. 하지만 이러한 방식은 비선형 방정식들의 해를 풀어야 한다는 단점을 지니고 있으며 관측하기 어려운 NLOS에서 얻은 TDOA를 사용할 경우 결과값이 현저하게 나빠지는 문제점을 지니고 있다.

\*딥러닝 특히 LSTM을 활용한 사전 연구 소개

이러한 문제점을 해결하고자 현재 딥러닝을 활용한 여러 측위 시스템이 개발되고 있다. 특히 신호의 연속성을 살려 시계열 모델(Time Series Model)들이 자주 사용되고 있다. 예를 들어 GCC-PHAT을 사용해 steering vector로 분리한 UWB 신호를 BLSTM을 이용해 noise가 제거된 TDOA 값을 구하는 연구가 있었다[Ro,,] 또한, UWB 신호에서 cross correlation을 이용한 관측된 값에서 신호들 간 분리 작업을 통해 TDOA를 구한 뒤, LSTM 모델을 사용하여 태그의 위치를 측위하는 연구도 존재한다[UWB,,,]. 그리고 UWB 신호에 포함된 noise를 제거하기 위해 학습된 LSTM 모델을 이용해 정확한 TDOA 값을 찾은 후, weighted KNN 모델로 타겟의 위치 좌표를 찾아내는 연구 또한 존재한다[wknn,,].

\*딥러닝 특히 CNN 을 활용한 사전 연구 소개

이 밖에도 UWB 분야에서 CNN 모델 또한 자주 사용되고 있다. 예를 들어 [CNN,,,]는 CIR 값을 입력 데이터로 하여 물체의 위치를 찾아내는 CNN 모델을 개발한 연구이다. 또한, 웨이블릿 변환(Wavelet Transform)을 통해 신호들을 이미지로 변환한 뒤, CNN 모델을 통해 TDOA 값을 찾는 연구 또한 존재한다[Micro].

\*기존 딥러닝을 활용한 연구 문제점

하지만 이러한 사전 연구들을 활용하더라도 여전히 NLOS 경로에서 정확한 TDOA 관측하는 것은 여전히 힘든 일이다. 또한, TDOA 보정을 위해 여러 연산과 파이프 라인을 거치기에 실시간으로 실내에서 위치를 추적하는 모델로 적합하지 않다. 이를 보완하고자 TDOA image based target tracking (TITT) 모델이 등장하였다[].

\*TITT 모델 소개 및 의의 및 한계

TITT 모델은 매우 복잡한 실내 환경에서 타겟의 정확한 위치 파악에 특화된 end-to-end 모델이다. TITT 모델은 관측한 TDOA를 그대로 사용하는 특징이 있다. 모델의 구조는 TDOA에서 AWGN을 제거하기 위해 이미지로 변환한 뒤 이를 CNN 모델에 넣어 타겟의 위치 좌표를 추론하는 구조이다.

해당 모델을 검증한 결과, 간단한 프로세스 구조를 통해 빠른 연산 속도를 보였으며 위치 추론값의 오차에 대해 강건(Robust)하였다. 하지만 이러한 값은 오차가 많이 발생하는 NLOS 경로에 대해서만 한정적인 결과였다. 또한, 위치 좌표의 평균적인 오차는 단순한 FCNNs 모델과 비슷했으며 오히려 noise가 존재하지 않는 LOS 경로의 TDOA 값에 대해선 더 안좋은 물체 추적 능력을 보였다. 즉, 해당 연구는 이미지를 변환하는 과정을 통해 AWGN을 없애는 과정은 좋은 접근이였지만 최적화 하지 않는 프로세스로 인한 성능의 한계를 보여준 연구였다.

[blank line]

**2. Temperature scaling**

\* Confidence 설명 & 필요성

기계학습에서의 신뢰도(confidence)란 예측값이 얼마나 믿을 수 있는지 나타내는 척도이다. 예를 들어 의료, 자율 주행, 금융 등 중요한 의사 결정을 필요로 하는 모델에서 결과값이 얼마나 믿을만 한지 확인하는 방법이다. 최근 각광을 받고 있는 딥러닝을 활용한 분류 문제의 예측값은 softmax 함수를 적용해 예측값을 0과 1 사이로 mapping한 값이다. 하지만 딥러닝 예측값은 일반적으로 실제 확률과 일치하지 않고 overconfidence 하다는 연구 결과가 존재한다[On Calibration]. 이러한 경향은 의료, 금융, 자율 주행 등 중요한 판단을 해야 하는 상황에서 모델은 예측한 확률이 높지만 틀린 판단(High-confidence errors)을 할 가능성이 높다는 문제점을 내포하고 있다.

\* calibration설명

이러한 문제점을 해결하기 위해 모델 보정(Model calibration)이 필요로 한다. calibration 이란 모델이 예측한 확률이 likelihood of correctness 를 나타내도록 하는 것이다. 예를 들어 모델의 예측값이 0.8일때, 정확하게 10개 중 8개를 분류하도록 하는 것이다. 이는 곧 모델의 신뢰도와 정확도가 일치하는 상황이며 분류 모델이 실제 확률을 반영하고 있다는 뜻이다.

\* temperature scaling 설명

calibration 방법 중에는 모델의 예측 확률로부터 calibrated probability를 구하는 temperature scaling 방법이 존재한다. temperature scaling은 기존에 많이 사용되는 platt scaling(softmax에 들어가기 전의 마지막 결과값을 σ(azi+b)에 넣어 변환)을 확장시킨 방법이다. 해당 방식은 K개의 label 이 있는 다중 분류 문제일 경우, 양수의 스칼라 값 temperature T로 logit vector z를 나눈 뒤 softmax 를 통과시킨다. 이를 수식으로 나타내면 다음과 같다.



여기서 양의 값으로 나눠주기 때문에 argmax를 바꾸지 않아 classification error에 영향을 주지 않는다. 즉, T는 모델의 정확도에는 영향을 주지 않고, calibration 에만 영향을 미친다.T의 장점으로는 다른 scaling에 비해 간단하다. 또한, softmax가 exponential 하기 때문에, 큰 값에 민감하게 반응하며, 같은 방식으로 조금만 그 값을 줄여도 그 값이 기하급수적으로 감소한다. 따라서 해당 방식은 vector norm을 줄이고, over-confidence를 줄이는 데에 효과적이다.

blank line]

**3. Convolutional LSTM**

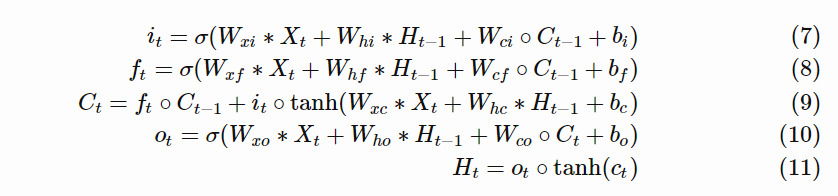
[blank line]

\* LSTM 설명

LSTM은 RNN 모델 중 한가지이며 Vanilla RNN의 최대 취약점인 장기간 데이터에 대한 학습 능력 저하인 장기 의존성 문제(the problem of Long-Term Dependencies)와 기울기 손실 현상(Vanishing gradient problem)을 해결한 모델이다. 하지만 컴퓨터 비전 분야에서는 입력 데이터의 구조를 변환하는 과정 때문에 공간적 특성을 사용하지 못하는 치명적인 단점을 지니고 있다.

\* Convolutional-LSTM 설명

Convolutional-LSTM은 앞서 진술한 단점을 보완한 모델이다. 해당 모델은 Fully Connected LSTM 모델에서 발전한 모델[generating]이다. Convolutional-LSTM의 자세한 연산 과정은 다음과 같다.



위의 식을 비교했을 때, 기존의 FC-LSTM 모델과 상당히 유사하다[fc-lstm 맨처음]. 하지만 해당 식은 다음 2가지의 결적인 차이를 가지고 있다. 1) 해당 모델은 입력-망각-출력 게이트(i,f,o)와 셀 입력, 출력 상태(X,C,H)가 3차원 벡터로 계산된다. 이를 통해 3차원 채널 데이터의 손실없이 시퀀스 데이터의 학습이 가능하다. 2) FC-LSTM에서 행렬곱이 사용되는 부분은 모두 합성곱으로 대체되었다. 이는 각 셀에서 모든 weight의 수가 FC-LSTM보다 줄어들게 되며, 전체적으로 모델이 학습할 weight 수를 크게 줄일 수 있게 된다.

한마디로 표현하자면 Convolutional LSTM은 FC-LSTM 내부 계산에 Convolution 과정으로 대체한 것이다. 이러한 과정을 통해 시간적, 공간적 상관성(Spatiotemporal correlation)을 동시에 학습할 수 있다는 큰 장점이 있다. 그 결과 컴퓨터 비전 분야에서는 기존 LSTM, FC-LSTM 모델보다 우수한 성능을 갖는다[temp21].

**3. Proposed UWB Localization Approach**

**3.1. Model Structure Overview**

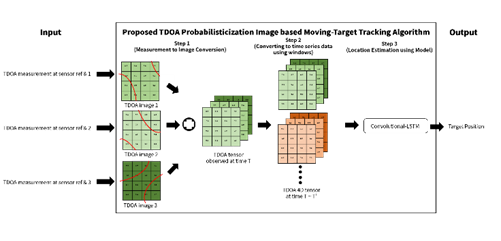


Figure 1 - System architecture for our propose TDOA Probabilisticization Image based moving-Target Tracking (TPITT ) model

\* 제안하는 모델 구조 설명

본 절에서는 이번 논문에서 제안하는 모델의 구조를 설명한다. 제안하는 TPITT 모델은 Figure 1과 같으며 4개의 UWB 앵커와 1개의 태그를 사용해 얻은 TDOA를 입력 값으로 한다. TPITT 모델은 앞서 설명한 TITT의 전반적인 프로세스를 계승하지만 이미지로 표현하는 과정과 학습하는 딥러닝 모델을 보완하였다. 모델의 프로세스는 다음과 같다. 1) 입력된 TDOA 값들은 해당 지역에 태그가 존재할 확률이 적혀있는 이미지로 변환하는 과정을 거친다. 이렇게 변환한 이미지들은 합쳐져 특정 시간 t에 각각의 앵커에서 관측한 TDOA 3D Tensor로 저장된다. 2) TDOA 3D Tensor는 시계열 데이터로 전환하기 위해 고정된 창(windows)을 이용해 4차원 구조의 데이터(TDOA 4D Tensor)로 전환한다. 3) 타겟의 위치 정보가 적혀있는 이미지 데이터(TDOA 4D Tensor)를 LSTM 안에 CNN 모델을 삽입한 Convolutional – LSTM 신경망 모델을 사용하여 이동하는 물체의 위치 좌표를 예측한다.

**3.2. TDOA to Image Conversion**

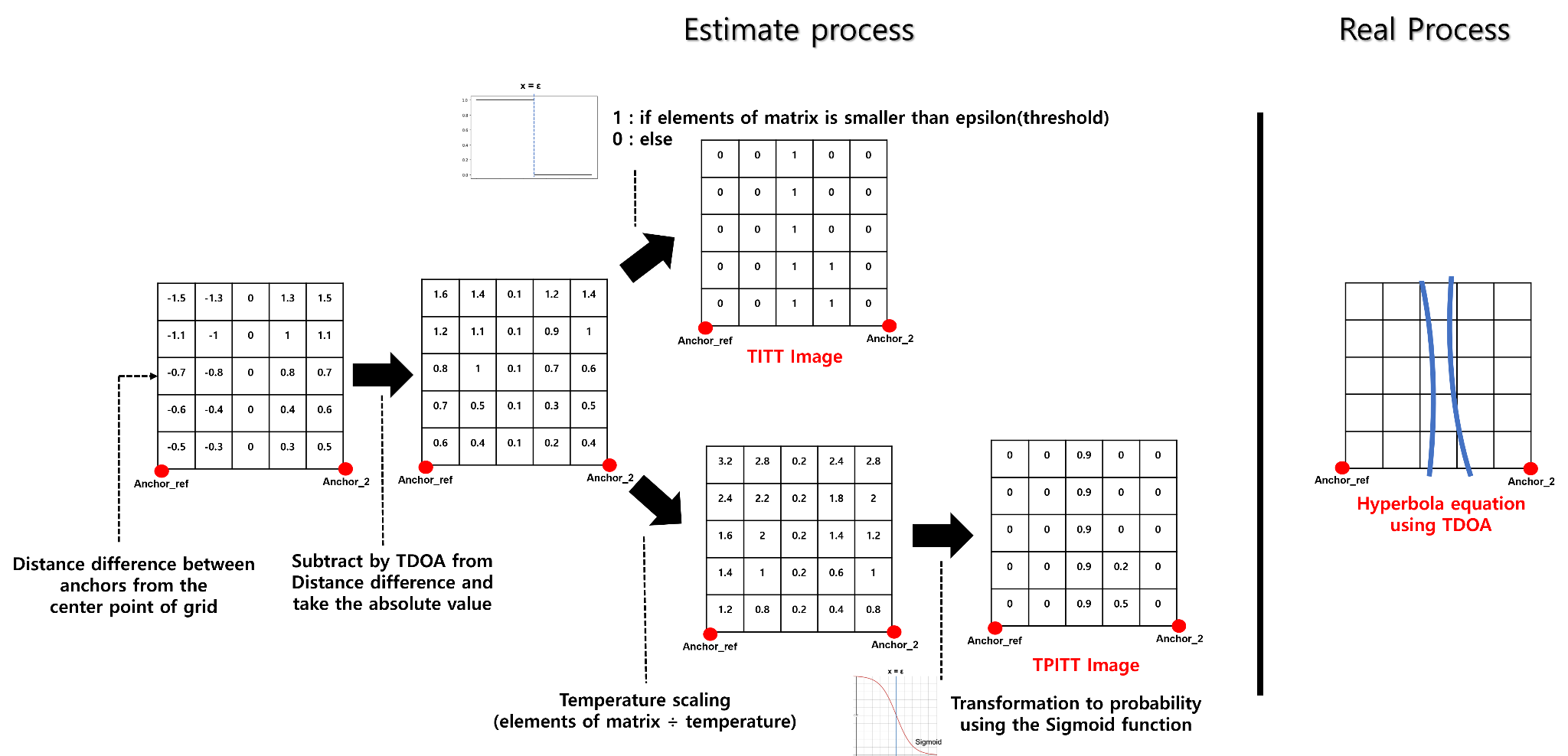
****

Figure 2 -Process Difference between A and B.

\*제안하는 이미지 전환 방법 아이디어 설명

사전에 진행되었던 TITT 연구와 마찬가지로 TDOA 값을 이미지로 변환하여 TDOA에 값에 포함되어 있던 AWGN의 영향을 낮추는 작업을 진행한다. 이미지로 변환하기 위해선 먼저 앵커 2개의 공간 좌표와 TDOA, 이미지 크기, 격자 크기, 최소 거리 차이 상수(ε), temperature(T)를 표현하는 하이퍼 파라미터를 필요로 한다.

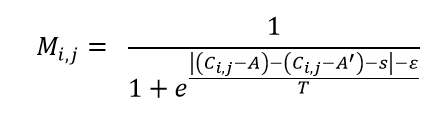
하이퍼 파라미터의 용도에 대해 자세하게 설명하자면, 격자 크기는 주어진 공간을 얼마나 자세하게 나눌 것인지 결정하는 변수이다. 또한, 최소 거리 차이 상수는 앵커와 격자 중점의 거리 차이와 TDOA와 유사하다고 비교할 때 필요한 작은 양수이며 temperature는 softmax 함수의 출력값을 soften 하게 조절해주는 양수이다.

하이퍼 파라미터 설정을 완료했다면, 0으로 채워진 가로, 세로가 이미지 크기인 행렬 M을 그린다. 이때 행렬의 원소는 격자의 중점과 기준 앵커(anchor\_ref)의 거리를 격자의 중점과 나머지 앵커(anchor\_2)의 거리로 뺀 값이다. 이때, 격자의 중점 좌표 C는 기준 앵커의 좌표를 A라 정의하면 다음과 같다.



위 식을 통해 구한 행렬 M의 각 원소에 TDOA 값을 뺀 뒤, 절대 함수를 적용한다. 이렇게 구한 행렬 M의 각 원소를 최소 거리 차이 ε 과 대소차이를 비교하여 각 원소를 작거나 같으면 1, 크면 0으로 채워 넣으면 TITT 방식의 TDOA 이미지 전환 방법이다.

필자가 제안하는 TPITT 이미지 전환 방법은 앞서 구한 행렬 M의 원소에 temperature scaling을 진행하여 행렬M을 calibration 하는 것이다. 앞서 만들어진 격자 각각의 중점 좌표를 C, TDOA 값을 s, 기준 앵커의 좌표를 A, 나머지 앵커의 좌표를 A’, 최소 거리 차이를 ε, Temperature를 T라고 가정하면 TPITT TDOA 이미지 변환 전체 프로세스를 다음 수식과 같이 나타낼 수 있다.



해당 수식을 통해 각 격자에 물체가 존재할 확률을 행렬 M의 원소에 표현한 TDOA 이미지를 구할 수 있게 된다. 이러한 과정은 AWGN 신호를 구분하기 위해 일반적으로 사용되는 Time-Frequency Masking 방법[]처럼 특정 공간에서의 타겟의 위치를 세밀하게 표현이 가능하도록 하였다. 또한, TITT 모델의 epsilon을 기준으로 하는 구간 함수 대신, temperature scaling과 softmax 함수를 이용해 연속함수로 바꾸는 과정이다. 이와 같은 방식으로 각각의 앵커에서 특정 시간 t에 측정한 TDOA 값을 이미지로 변환하고 합쳐져 3개의 채널을 가진 Tensor로 저장한다.

\*제안하는 이미지 전환 방법에 필요한 하이퍼 파라미터 에 대한 이해

이때, 여러번의 실험을 통해 하이퍼 파라미터는

1) grid size는 작을수록 좋지만 계산 cost가 기하급수적으로 상승하므로 적절한 trade-off 과정이 필요하다.

2) 여러 번의 실험 결과, 최소 거리 차이 ε은 grid size의 절반과 비슷할 때 좋은 성능을 보인다.

라는 상관 관계를 알 수 있었다.

**3.3. 4D tensor construction with time window**

\*시계열 데이터 변환 필요성 및 전환 방법

UWB 센서를 이용해 실시간으로 데이터가 앞선 이미지로 변환하는 과정을 마치고 3차원의 배열(가로, 세로, 채널)로 수집된다. 3차원 형태의 데이터를 Convolutional – LSTM에서 Convolutional filter의 입력 값으로 사용하기 위해 4차원의 시계열 데이터 형태로 변환하는 과정이 필요하다.

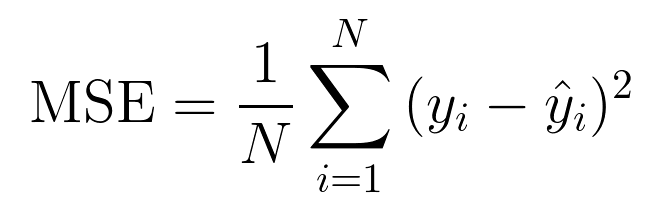
본 논문에서는 이러한 변환을 위해 윈도우 값을 4로 설정한다. 즉, 4초간의 타겟의 움직임을 고려해 최종적으로 물체의 위치를 파악하고자 하였다. 그 결과 최종적으로 모델의 학습에 사용되는 데이터의 형태는 (윈도우, 가로, 세로, 채널)이다.

**3.4. Estimation using Convolutional-LSTM**

\*모델 구조 및 입-출력값 설정 & 설명

본 실험에서는 실시간으로 이동하는 물체의 특성에 맞춰 Convolution-LSTM 모델을 사용한다. Convolution-LSTM 모델은 CNN을 통해 만들어지는 필터로 구별되는 공간적 정보와 LSTM 구조를 통해 시간 정보를 반영하여 각 step에서 시계열 정보를 학습할 수 있다. 즉, 해당 모델을 통해 시간에 공간적(Spatial), 시간적(Temporal) 변화 파악이 가능하다. 이를 활용해 매 초 속도와 방향이 변하는 물체의 성질을 반영하여 물체의 공간 좌표에 대해 regression이 가능하다. 모델의 구조는 아래 그림과 같다.

본 모델은 4차원의 시계열 TDOA 이미지 데이터를 입력으로 하고 물체의 공간 좌표를 출력값으로 한다. 또한, 앞서 언급한대로 회귀 모델에 속하므로 loss 함수로 Mean Square Error(MSE)를 사용한다. 이때, 다음 식에서 N은 데이터셋의 개수, y는 물체의 실제 공간 좌표, ŷ은 모델이 예측한 물체의 공간 좌표이다.



**4. Experiment**

**4.1. Environment setting & data generation**

\*컴퓨터 시뮬레이션을 하기 위한 실험 세팅 정의

제안하는 모델의 성능 검증을 위해 컴퓨터 모의실험을 수행한다. 본 실험은 단일 물체에 대해 위치를 추정하기 위해 크기는 10m \* 10m의 정사각형 공간에 각 꼭지점에 4개의 센서가 위치하고 물체의 움직임을 추적하고 있으며, 그 중 하나가 기준 센서로 선택된다고 가정한다. 채널 모델은 IEEE 802.15.4a의 사무실 내 환경을 가정하며 “Bitcraze” 사의 “loco positioning system” 프로그램을 이용하여 거리에 따른 시간 지연과 잡음을 더하여 수신 신호를 만든다. 각 센서의 TOA를 기반으로, TDOA는 기준 센서와 나머지 센서 간의 TOA 차이로 중앙 컴퓨팅 유닛(CCU)에서 계산된다. TDOA 측정은 millisecond 단위로 이루어지며 각 센서와 CCU 사이의 통신 지연은 없다고 무시한다.

다양한 환경을 가정하기 위해 환경에 따라 각각 다른 측정 오차를 적용한다. 이때, 측정 오차는 환경의 컨디션에 따라 정의가 가능한 분산이 AWGN으로 모델링된다. 본 실험에서는 1) 장애물이 거의 없는 환경은 평균 0, 표준 편차가 0.2, 2) 장애물이 보통인 환경은 평균 0, 표준 편차가 1, 3) 장애물이 많은 환경 : 평균 0, 표준 편차가 2, 인 AWGN이 실제 TDOA에 더해진다.

이렇게 실험 환경 세팅이 완료되었다면 물체의 움직임 데이터를 생성한다. 실험은 다양한 상황에 대해 범용적으로 작동하는 모델을 찾아야 하기 때문에 다양한 움직임을 보이는 물체의 데이터를 생성한다. 이때, 데이터는 물체가 지그재그, 랜덤, 회전 운동을 한다고 가정한다. 물체는 가속도 운동을 진행하며 물체의 가속도는 표준정규분포에서 얻은 값을 통해 매초 업데이트 된다. 이때, 속도의 최대값은 인간의 평균 달리기 속도인 5m/s이다.

물체의 운동 조건이 정해졌다면 앞서 가정한 환경에 맞춰 학습용 데이터인 랜덤으로 운동하는 물체의 데이터 16,000개를 생성한다. 그런 뒤, 모델 평가용 데이터로 전체 공간에서 지그재그, 랜덤, 회전 운동하는 물체의 데이터 각각 4,000개, 기준 앵커 주변 공간(반경 2m)에서만 지그재그, 랜덤, 회전 운동하는 물체의 데이터 각각 4,000개를 생성한다.즉, 환경 하나에 대해 1개의 학습 데이터셋(전체 공간 랜덤)가 존재하고 6개의 평가 데이터셋(전체 공간 랜덤 / 지그재그 / 회전 & 국소 공간랜덤 / 지그재그 / 회전)가 존재한다.

**4.2.Model hyper Parameter**

TPITT Convolutional LSTM 모델을 검증하기 위해 비교군 모델은 총 4개가 존재한다.(FCNNs, TPITT CNN, TITT CNN, TITT Convolutional LSTM) 이때, CNN 모델의 구조는 Convolutional LSTM 층이 일반 Convolution 층으로 바뀐 것을 제외하고 모든 구조가 동일하다. FCNNs 모델은 iput이 TDOA 3개이며 hidden layer는 16, 8 이고 output layer는 2인 구조를 가지고 있다. 딥러닝 모델을 학습하는데 필요로 하는 하이퍼 파라미터는 아래의 표와 같이 5개 모델에 모두 똑같이 적용한다. 이밖에 6번 epoch 이상 모델의 성능이 개선되지 않으면 learning rate의 80%로 감소하며, 8번 이상이라면 학습을 종료하는 callback 함수를 사용한다.

| Parameter | Value |
| --- | --- |
| 활성화 함수 | Relu |
| Optimizer 함수 | Adam |
| 학습률 | 0.001 |
| Loss 함수 | MSE |
| Drop out | 0.6 |
| 미니 배치 크기 | 20 |
| Epoch | 30 |

표 1 신경망 파라미터 설정

**4.3. Model comparison**

사전에 정의한 조건대로 3개의 환경에서 5개의 모델들을 학습하고 6개의 데이터셋으로 검증한 결과는 다음 표와 그림과 같다.

fig 5를 보면 더욱 자세히 알 수 있다. fig 5는 noise가 매우 많은 공간에서 원형 운동과 랜덤 운동을 하고 있는 물체의 실제 좌표와 예측 좌표를 좌표 평면에 나타낸 것이다. 전체 공간에서 FCNNs 모델은 Conv-lstm과 거의 비슷한 좌표를 예측하고 있다. 하지만 앵커 주변의 공간에서는 앞선 예측과 다르게 엉뚱한 곳을 예측하고 있다.

또한, fig 6를 살펴보면 noise가 거의 없는 공간에서는 TITT와 TPITT는 거의 비슷한 성능을 보인다. 하지만 점차 노이즈가 증가할수록 TITT와 TPITT의 성능 차이가 더욱 커지는 경향을 보이고 있다.

본 실험은 이미지 변환을 통해 AWGN을 줄이고 다양한 타겟과 환경에서 보편적으로 사용할 수 있는 위치 추적 방법을 찾고자 TPITT 모델을 제안하였다. 그 결과 먼저, TDOA 이미지 기반의 모델이 FCNNs 모델에 비해 앵커 주변의 구석진 공간에서 또한 안정된 성능을 보였다. 즉, 이미지 기반의 모델은 숫자 기반의 모델과 다르게 보편적인 상황만 학습을 진행하여도 특이하고 다양한 상황에서도 강건하게 섬세하게 예측을 한다.

또한, 본 논문에서 제안하는 TPITT 모델은 AWGN이 증가할수록 TITT 모델에 비해 우수한 성능을 보인다. 즉, TPITT의 확률화 과정은 이미지 변환 방식의 주요 목적인 AWGN을 제거하는 역할을 더 잘 수행한다는 사실을 알 수 있다.

마지막으로 다른 모델에 비해 Convolutional-LSTM 모델이 전반적으로 가장 좋은 성능을 보인다. 이는 모델이 물체의 운동성을 파악하고 학습하여 움직이는 물체의 위치를 잘 파악할 수 있었다. 즉, 움직이는 물체를 추적하는 실생활의 용도를 고려했을 때, 다른 모델들에 비해 Convolutional-LSTM은 범용적으로 사용할 수 있는 모델이다.

이를 종합하자면 TPITT 모델은 기존의 연구 대비 복잡한 환경에서의 성능이 향상되었으며 어떤 물체와 환경이든 강건하게 성능을 보이는 모델이다. 이를 활용해 차후 다양한 환경에서 Indoor tracking을 진행할 수 있을 것이라 예상된다.

**5. Conclusion**