

출품번호
1447

제70회 전국과학전람회

**전략 기반 군중 유동 역학 모델
제시 및 현상 분석 적용**

2024.08.04.

출품학생	고민재, 김성훈
지도교원	문기보
구 분	학생부
출품부문	산업 및 에너지(SW·IT 융합 분야 포함)

1. 서론	4
가. 제작 동기와 목적	4
나. 선행연구 및 관련 이론	5
1) 군중 유동 역학(Crowd Flow Dynamics, CFD)	5
2) 이산요소법(Discrete Element Method, DEM)	6
3) Verlet Integration	7
4) Crowd Localization	8
5) Inverse Perspective Mapping(IPM)	9
6) 칼만 필터(Kalman Filter)	11
7) Dynamic Time Warping(DTW)	12
8) 베이지안 최적화(Bayesian Optimization)	13
2. 연구방법 및 이론	13
가. 도구 및 재료	13
나. 제작 과정	14
1) 전략 기반 CFD 시뮬레이션 제작	14
가) 물리 모델의 구현	14
나) 전략의 반영	15
다) 다양한 상황에서 시뮬레이션 테스트	17
2) 현실 군중 데이터의 변환	17
가) IIM 모델의 적용과 Crowd Localization	17
나) Homography Matrix H의 계산과 IPM 적용	17
다) 개별 개체 분리 및 칼만 필터 적용	18
라) 다양한 현실 군중 데이터로의 테스트	19
3) 데이터 비교를 통한 전략 최적화 방안 제시	19
가) 움직임 유사도 측정 방법 고안	19
나) 전략의 파라미터 최적화	19
4) 예시 시나리오 적용: 도쿄, 시부야 스크램블 교차로의 횡단보도	19
3. 결과	20
가. DEM을 적용한 전략 기반 시뮬레이션 제작	20
나. 현실 군중 데이터의 변환	22
1) IIM 모델의 적용과 Crowd Localization	22
2) 다양한 현실 군중 데이터로의 테스트	22
가) Worst Case	22
나) Best Case	23
다. 예시 시나리오 적용: 도쿄, 시부야 횡단 상황	24
1) 현실 군중 데이터 변환 결과	24
2) 시뮬레이션 최적화	25
4. 고찰 및 결론	26
5. 제언	26
6. 참고 문헌	27

1. 서론

가. 제작 동기와 목적

일반적으로 시뮬레이션 연구는 시스템의 복잡한 동작을 모델링하고 예측하기 위해 다양한 수학적 및 계산적 기법을 활용한 연구를 진행하는 분야를 의미한다. 시뮬레이션 연구는 정밀한 수치 해석을 통해 예측 정확성을 높이는 것을 목표로 한다. 예를 들어 기후 모델링, 유체 역학, 구조 해석 등에서는 물리적 법칙을 기반으로 한 방정식을 해결하여 시스템의 미래 상태를 예측한다. 이러한 모델은 시스템이 특정 물리적 법칙 등에 의해 결정될 때 효과적이다.

군중 유동 역학(Crowd Flow Dynamics, CFD)은 특정한 목적성이나 방향성을 가지는 군중의 움직임이 유체 역학적인 패턴을 떤다는 사실에 기인해 물리 모델을 적용하여 분석하는 분야이다. 2022년 ‘이태원 참사’라고 불리는 ‘이태원 암사 사고’가 발생했었다. 이는 대표적인 군중 난류에 의한 사고로 밀집 군중이 좁은 통로에 집중되면서 발생한 안타까운 사건이었다. 현재도 많은 콘서트장, 행사장에서 밀집 군중의 움직임에 관련된 사고들이 발생하고 있고, 이는 대표적인 CFD 분야의 핵심 주제이다. 군중의 움직임을 분석하는 만큼 CFD 또한 시뮬레이션을 자주 활용한다. 그러나 CFD 시뮬레이션은 일반적인 시뮬레이션보다 몇 가지 추가적인 어려움을 가지고 있다. 첫째, 군중은 수많은 개별 개체들로 구성되어 있으며, 이들은 각기 다른 의사 결정 과정을 통해 움직인다. 이 과정에서 각 개체의 행동은 주변 환경과 상호작용하면서 복잡한 패턴을 형성하게 된다. 둘째, 군중의 움직임은 단순한 물리적 법칙으로 설명할 수 없는 사회적, 심리적 요인들에 의해 영향을 받는다. 예를 들어, 공황 상태에서의 군중은 평상시와 매우 다르며 예측하기 어려운 패턴을 띠게 된다. 이러한 이유로 기존의 CFD 연구들은 임의로 설정한 인간의 움직임이 실제 현상을 설명한다는 것을 확인하거나, 시뮬레이션의 타당성을 입증하는 것은 뒤로 한채 다양한 상황에 적용하는 것에 초점을 맞춘다. 즉, CFD 시뮬레이션이 예측에 적용되기엔 한계가 있는 것이다.

그래서 본 연구에서는 CFD 시뮬레이션의 타당성을 현실 군중 데이터와의 비교를 통해서 입증하는 동시에 높은 활용성을 제공할 수 있는 ‘전략 기반 군중 유동 모델’ 프레임워크를 제안했다. 전략은 위에서 언급한 물리적 법칙으로 설명되지 않는 군중의 상호작용을 의미한다. 전략을 유동적으로 변화시킬 수 있는 구조를 설계하여 최적화를 진행하는 것이 핵심 아이디어이다. 구체적으로 이 연구에서는 다음의 과정을 포함하여 프레임워크를 구성했다.

- 1. 전략 기반 CFD 시뮬레이션 제작:** 개별 개체의 의사 결정 과정을 반영하여, 군중의 움직임을 보다 현실적으로 재현한다. 이를 위해 이산요소법(Discrete Element Method, DEM)을 적용하여 개체 간 상호작용을 정밀하게 모델링한다.
- 2. 현실 군중 데이터의 변환:** 이미지 상에서 군중의 위치를 얻어내는 Crowd Localization과 이를 재구성하는 Inverse Perspective Mapping(IPM)을 사용하여, 현실에서 수집된 군중 데이터를 시뮬레이션과 대응시킬 수 있도록 한다.
- 3. 데이터 비교를 통한 전략 최적화 방안 제시:** Dynamic Time Warping(DTW)를 사용하여, 시뮬레이션 결과와 현실 군중 데이터를 비교하고, 이를 기반으로 시뮬레이션 전략의 파라미터를 최적화하는 방법을 사용한다.

이 프레임워크를 통해 CFD 시뮬레이션의 타당성을 입증함으로써 시뮬레이션을 예측에 적용하며, CFD가 다양한 분야로 확장되어 활용성을 높일 수 있도록 하는 것이 이번 연구의 궁극적인 목표이다.

나. 선행연구 및 관련 이론

1) 군중 유동 역학(Crowd Flow Dynamics, CFD)

군중의 운동은 거시적인 관점에서 밀집도에 따라 불규칙한 운동을 하게 되는데, 이에 대해 물리 모델을 제시해 시뮬레이션을 진행하며 다양한 현상을 분석하는 연구 분야를 CFD라고 한다. CFD 연구는 크게 시뮬레이션을 구현하는 방식, 목적에 따라 종류가 나뉜다.

먼저, 시뮬레이션을 구현하는 방식에 따라 움직임을 이산적으로 묘사하고 특정 환경에 취하는 전략에 대해 집중적으로 탐구하는 Discrete Model 방식과 운동방정식을 통해 움직임을 연속적으로 묘사하고 현실적인 현실 묘사에 집중하는 Continuous Model 방식이 있다. Discrete Model은 포텐셜을 정의하거나 인간 상호작용과 관련된 별도의 규칙을 만드는 식의 방식을 통해 인간의 전략을 모델링 한다. Discrete Model을 통한 CFD 구현 연구에는 남성원의 포텐셜 유적선법 프로그램인 P*FLOW를 사용해 환경에 대한 포텐셜을 정의하고 이에 따라 전략을 취하는 연구[1]와 서로 다른 방향을 향하고자 하는 다수의 집단이 존재하는 상황에서 도로와 집단의 전략과 정체되는 정도의 상관관계와 관련된 연구[2,3]가 존재한다. Discrete Model은 모델의 파라미터 설계가 편리해 환경에 따른 보행자의 경로 선택을 이해하고 적용하는 것이 쉽다. 또한 시간복잡도적인 측면에서 유리함이 크다. 하지만, 불연속적으로 인간의 움직임을 묘사해 현실과의 괴리가 커 환경을 정확히 묘사하는 데에 어려움이 있다는 근본적인 한계점이 있다. Fig 1은 Discrete Model의 구현 예시를 나타낸 사진이다.

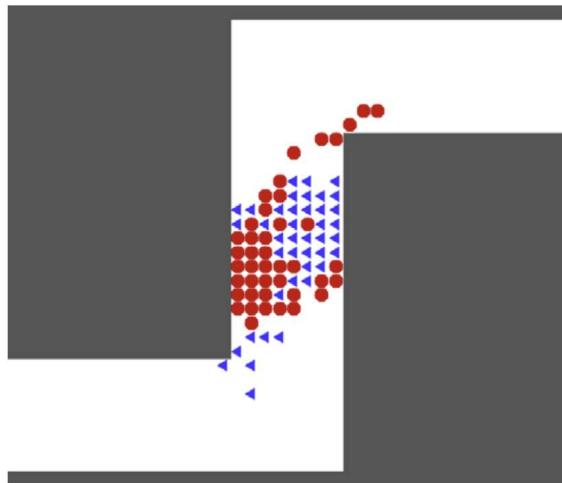


Fig 1. Discrete Model의 구현[3]

Continuous Model을 통한 CFD 연구는 일반적으로 이산요소법(Discrete Element Method, DEM)을 사용하여 인간을 이산 요소로 모델링해 힘을 계산하고 상호작용을 구현하는 방식이 주로 사용된다. CFD 시뮬레이션에 DEM을 적용한 연구는 개개인의 전략보다는 연속적인 움직임을 구현함으로써 현실과 유사한 환경 조건에 따른 군중의 유동 변화를 관찰하기 위한 것이 대부분이다. 대표적으로 사람을 3개의 원으로 모델링하고, 이를 DEM을 통해서 다양한 환경 상황 속에서 대피가 가장 빠르게 이루어지는 경우와 해당 경우의 특징에 대해서 조사한 Langston, Masling, Asmard의 연구[4]가 있다. 본 연구에서 또한 현실과 유사한 시뮬레이션 구현이 중요했기에, 시간복잡도가 조금 복잡한 대신 정확한 Continuous Model을 적용하였다.

다음으로 목적에 따라 크게 두 가지 갈래로 나뉜다. 첫째, 제시한 시뮬레이션을 통해 군중 난류나 정체 상황, 다양한 환경 조건에서 어떤 전략을 취하는 것이 군중의 흐름에 유리하게 작

용할지에 관한 연구가 있다. 대부분의 Discrete Model을 활용한 연구와 추가적으로 정체 상황에서 군중의 전략을 탐구한 Xu, M. L.의 연구[5] 등이 이에 해당한다. 둘째, 현실성이 있는 움직임을 구현하고, 이를 통해 이미 발생한 사건을 분석하는 연구가 있다. 이 연구들은 실제 인간과 유사한 모델을 구현하기 위해 전략을 탐색하는 것에 초점을 맞추고 있다. 전략의 조정에 어려움이 있는 대부분의 Continuous Model 관련 연구들이 이에 해당한다.

인간과 유사한 행동을 구현하기 위해 전략을 탐색하는 연구에는 다양한 방식이 활용된다. 먼저, 직접 전략을 구현하는 방식이 있다. 이외의 방식으로는 유전 알고리즘이나 강화학습을 사용해 군중이 서로 충돌하지 않고 목표물에 도달할 수 있도록 학습하는 방식이 있다. 유전 알고리즘을 CFD에 적용한 Sagredo-Olivenza의 연구[6]가 있다. 이들은 3D 환경 시뮬레이션에서 적은 연산으로 복합적인 전략이 고려되는 군중 시뮬레이션을 진행하기 위해서 유전 알고리즘을 적용했다. Deep RL[7]을 CFD에 적용해 State를 받아 Action을 생성하는 Policy 함수를 전략으로 학습하는 이재동과 원정담의 연구[8]가 있다. 이 연구는 주변 군중의 상황, 현재 이동 속도 등을 State로 하여 이를 기반으로 속도와 각속도를 Action으로 얻어낸다. 해당 연구는 강화학습의 Reward를 목표물에 가까워지는 정도 그리고 다른 객체와의 충돌 여부, 매끈한 움직임으로 설정하여 학습을 진행했다. Fig 2는 Agent의 State를 얻어내는 방식을 모식화한 그림이다. 본 연구에서는 이 연구에서 전략을 구현한 아이디어에서 착안하여 주변의 정보를 상태로 받아 행동을 출력하는 함수의 형태로 전략을 유동적으로 변화시킬 수 있도록 구현했다.

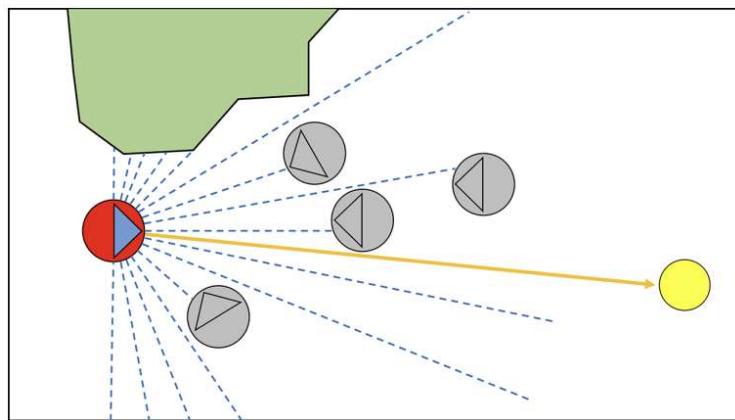


Fig 2. Deep RL의 CFD 시뮬레이션 적용[8]

본 연구에서는 기존의 대표적인 CFD 연구의 접근과 전혀 다른 접근을 시도한다. 기존의 CFD 시뮬레이션이 현실적인 움직임, 상황을 구현하기 위해 다양한 방식을 적용하였으나, 이에 대한 평가는 단순한 직관이나 추측에 근거할 뿐이었다. 즉, 현실성을 입증할만한 마땅한 수단이 없었던 것이다. 본 연구는 전략을 유동적으로 변화시킬 수 있는 전략 기반의 시뮬레이션을 구성하고, 현실성을 수치화하여 전략을 최적화하는 방식을 제안함으로써 기존 CFD 연구 방식과 다른 타당성이 높은 접근을 시도한다는 점에서 차별점이 있다.

2) 이산요소법(Discrete Element Method, DEM)

Continuous Model을 사용하는 CFD 시뮬레이션은 기본적으로 개별 개체를 입자로 모델링하여 물리적인 법칙을 적용한다. 대표적으로 불연속적인 입자들의 상호작용과 움직임을 분석하기 위해서 입자 간 접촉 및 충돌을 이용하는 모델링 기법인 DEM을 사용한다[9]. 본 연구에서도 DEM을 적용하여 시뮬레이션을 제작했다. DEM은 크게 구형 입자 간 겹침을 허용하고 겹침 정도에 따른 반발력을 통한 방식인 Soft Sphere DEM과 겹침은 허용하지 않되, 반발계수를 통해

서 충돌 후 운동량의 변화를 구현하는 Hard Sphere DEM 방식으로 구분된다[10]. Soft Sphere DEM 방식은 접촉력이나 변형력을 빠르게 계산할 수 있어 본 연구에서는 Soft Sphere DEM을 사용하였다. Fig 3는 Soft Sphere DEM과 Hard Sphere DEM을 비교한 그림이다.

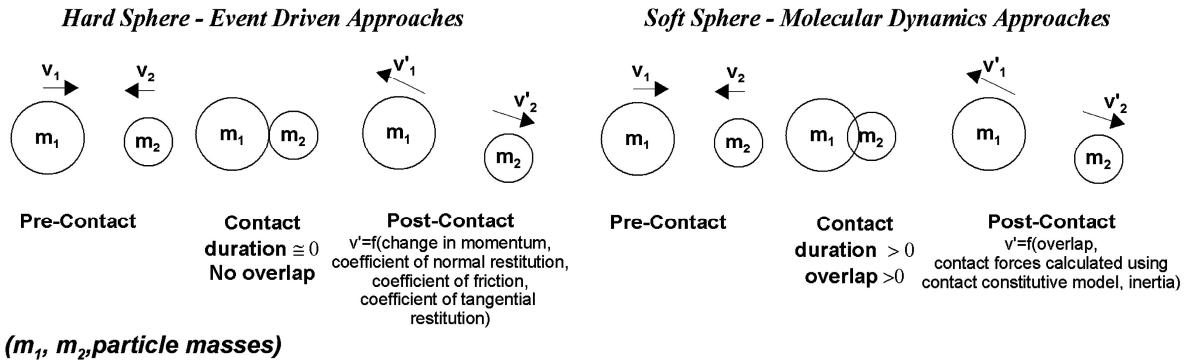


Fig 3. Hard Sphere DEM(좌)과 Soft Sphere DEM(우)[10]

모든 접촉한 입자 사이의 상호작용을 계산하는 DEM에서는 효율적으로 접촉한 입자들을 검출하는 알고리즘이 필요하다. 본 연구에서는 그리드에 따라 계산 영역을 분할하고 영역의 단위로 입자를 선택해 접촉 여부를 계산하는 Linked-list Cell Method(LCM)를 활용한다[11]. Fig 4는 LCM의 작동 방식을 나타낸 그림이다.

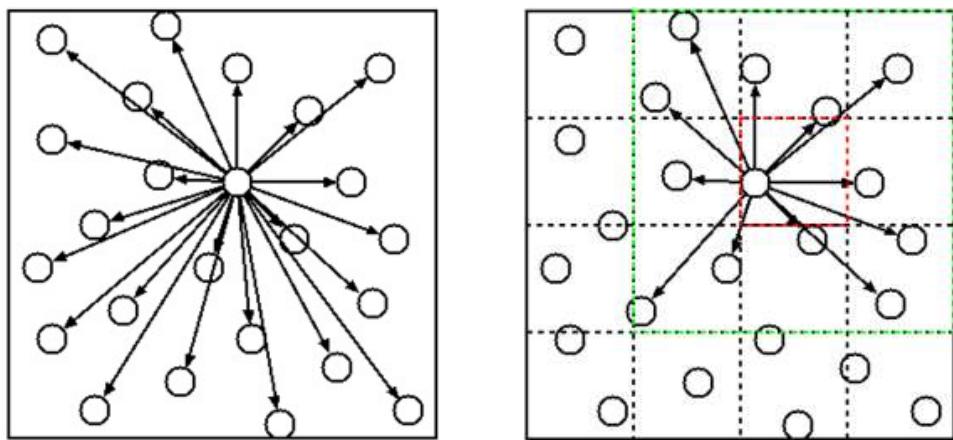


Fig 4. 모든 입자 쌍을 비교하는 경우(좌)와 LCM을 적용하여 그리드를 분할한 경우(우)[11]

3) Verlet Integration

Verlet Integration은 수치해석적 방법으로 뉴턴의 운동방정식을 적분하기 위해서 사용되는 방법이다[12]. 분자 역학 등의 다양한 시뮬레이션과 컴퓨터 그래픽 등에서 개별적인 입자의 궤적을 계산하는 데 자주 사용된다. Verlet Integration은 수치해석을 위해서 사용되는 간단한 오일러 방법에 비해 실행시간의 추가 없이 물리적 시스템에서 중요한 시스템과 높은 수치 안정성을 제공해준다. Verlet Integration은 오차가 4차 항으로 나타나 오차가 1차 항으로 나타나는 오일러 방법이나 2차 항으로 나타나는 Runge-Kutta 방법보다 운동방정식을 통한 입자의 궤적 계산을 정확하게 하는 것이 가능하다[13]. Verlet Integration은 시간 간격인 Δt 가 일정하지 않은 상황에서도 사용 가능한 방식 등으로 다양하게 확장 연구되었다. 본 연구에서는 가장 쉽게 구현 가능하여 통상적으로 사용되는 Velocity Verlet을 사용했다[14]. Velocity Verlet은 속도를 가속도와 함께 구하는 것을 통해서 Verlet Integration의 첫 번째 단계의 문제점인 초기값이 많이 요구된다는 점을 해결했다. Velocity Verlet은 다음 과정을 통해서 입자의 움직임을 Δt 간격으로 구해내게 된다.

먼저, 수식 (1)에 따라 Δt 의 절반에 해당하는 상황에서의 속도를 현재 가속도를 적산하여 구한다. 이후 구해진 속도를 바탕으로 수식 (2)에 따라 변위에 해당하는 항을 더하고, 달라진 변위에 따라서 포텐셜을 통해 계산한 가속도 항을 수식 (3)에 따라 속도에 최종적으로 더해서 속도 변화를 계산한다. Velocity Verlet을 통해서 본 연구에선 군중을 이루는 임의의 개체에 대해서 가속도를 더하는 과정을 반복함으로써 최종적인 움직임을 구현하였다.

4) Crowd Localization

Crowd Localization은 군중의 운동 분석을 위한 한 과정으로, 이미지로부터 군중의 각 개별 개체의 위치를 얻어내는 것을 의미한다. 이는 일반적으로 큰 군중의 수를 분석하는 Crowd Counting 분야와는 달리, 개체 단위로 인식하여 더 정밀한 군중 분석을 가능하게 하는 기술이다. 큰 군중 데이터는 라벨링 방법에 따라 사람의 수만 기록하는 Total Count Annotation, 점으로 사람 머리를 구분하는 Dot Annotation, 개별 개체의 Bounding Box를 그리는 Box Annotation 등으로 구분된다. Crowd Localization을 모델로 접근하기 위해선 Dot Annotation, Box Annotation 이상의 데이터가 요구되기 때문에, Crowd Counting보다 개발 진척이 다소 느린 분야이다. Fig 5는 Crowd Localization에서 사용하는 대표적인 라벨링 방식을 나타낸 것이다.

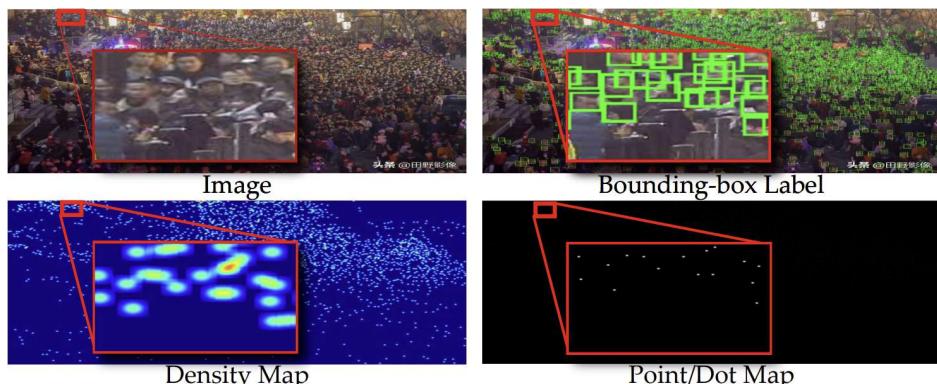


Fig 5. 군중 데이터의 다양한 라벨링 방식[15]

Crowd Localization과 Crowd Counting은 밀접하게 관련되어 있기 때문에, Crowd Localization을 이해하기 위해선 최근의 Crowd Counting 연구를 간략히 검토할 필요가 있다. 현재 대부분의 Crowd Localization 연구들은 기존의 수동적 특징 분석에서 벗어나 CNN 등의 딥러닝 모델을 이용한 접근 방식을 사용한다. 일부 방법은 픽셀 단위 또는 패치 단위로 군중 수의 밀도를 나타내는 Density Map을 회귀하는 데 중점을 둔다. 또는 직접적으로 Segmentation Map을 얻어내도록 학습을 진행하여 Crowd Counting을 진행하기도 한다. 그러나 이러한 방법들은 수 또는 대량적인 지역 밀도 예측에 초점을 두고 있기에 바로 Crowd Localization에 적용되기에 어려움이 있는 접근이다. 그래서 Crowd Localization은 다른 방법론으로 문제를 접근한다. 이는 크게 Detection-Based Method, Heuristic Method로 구분된다.

Detection-Based Method는 객체 인식을 통한 Bounding Box를 기반으로 위치를 예측한다. 대부분의 수동적인 이미지 특징 분석 모델이나 객체 인식용 CNN을 발전시켜 적용한 R-CNN, 개

별 객체 인식의 용이성을 더하기 위한 LSTM 레이어를 추가한 OverFeat 등이 Detection-Based Method에 속한다. 하지만 이들은 개별 개체가 겹치게 될 경우 성능이 크게 하락하며, 화질, 크기 변이에 크게 의존한다. 그 뿐만 아니라 모델을 통한 Bounding Box 검출은 정확한 위치 판정에 어려움을 준다. YOLO 모델 등의 Bounding Box 불확실성을 개선하기 위한 노력들이 있었으나, Detection-Based Method의 접근 방법 자체의 한계로 큰 진전이 있지 못했다[7].

Heuristic Method는 높은 화질의 데이터에 한정되어 Crowd Counting 모델을 통해 얻어낸 Density Map, Segmentation Map을 직접 이용하여 Crowd Localization을 진행하는 방법이다. 주로 가우시안 분포로 가정하여 근방에서 가장 높은 확실성을 나타내는 위치를 개별 개체의 머리로 특정한다거나, Density Map을 수동적으로 재구성하는 알고리즘을 개발하는 방향으로 접근하게 된다. 이런 접근은 Detection-Based Model보다는 위치 판정이 쉬우나, 안정적인 개별 개체 구분에 어려움이 있을 뿐만 아니라 마찬가지로 크기 변이에 취약하다. 이를 개선하기 위해서 Independent Instance Map(IIM)을 제안하여, Density Map으로부터 이를 얻어내기 위한 과정의 일환으로 Binarization Module(BM)을 추가한 연구[15]가 있다. Density Map의 일종인 Confidence Map에 Binarization을 적용하기 위한 국소적인 Threshold를 적용하는 Threshold Map을 얻어내는 레이어가 BM의 내부에 포함된다. 이 Threshold에 따라 Binarization을 적용하여 IIM을 얻어내게 된다. Fig 6는 이 모델의 구조를 나타낸 그림이다. Confidence Map을 위한 예측기로는 미리 학습된 VGG-16 + FPN 모델과 HRNet-W48이 활용되었다고 한다.

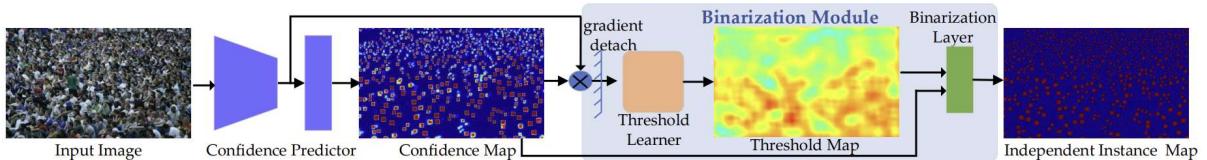


Fig 6. Binarization Module 추가를 통한 IIM 변환 Crowd Localization 프레임워크[15]

본 연구는 영상으로부터 현실 군중 데이터를 얻어내어 시뮬레이션을 적용할 수 있는 형태로 변환하는 과정이 프레임워크에 포함된다. 일반적인 CFD 연구로의 적용을 위해선 밀도가 높고, 크기 변이가 큰 군중의 데이터에도 이 데이터 변환 과정이 효과적이어야 한다. 그래서 본 연구에서는 NWPU-Crowd 데이터셋에서 가장 높은 성능을 보이는 State-of-the-Art 기술이자, 이미지에서 영상으로의 확장에 적합한 연산량을 지닌 IIM 모델을 적용했다.

5) Inverse Perspective Mapping(IPM)

Inverse Perspective Mapping(IPM)은 카메라로 촬영된 2D 이미지를 실제 좌표가 드러나는 Bird Eye View(BEV) 형태의 데이터로 변환하는 기술로, 자율 주행 시스템에서 주로 활용된다. IPM의 핵심은 원근 사영 과정에서 발생하는 원근 왜곡을 제거하여 2D 이미지에 담긴 일관된 3D 장면을 BEV 데이터로 재해석하여 표현하는 것이다. 본 연구에서는 현실 군중 데이터의 원근 왜곡 제거를 통해 공간 데이터를 시뮬레이션 환경에 적합하게 BEV로 얻어내는 과정이 필수적이기에 이를 사용하게 된다. IPM은 아래의 World 좌표계에서 Image로의 원근 사영 행렬의 계산으로부터 출발한다. 평행 이동의 계산 편의성을 위해서 동차 좌표계가 사용되었다.

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = K[R|t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = H \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \dots \quad (4)$$

X , Y , Z 는 World 좌표계에서의 공간 좌표를 의미한다. 이에 행렬 $[R|t]$ 를 취하여 회전과 평

행 이동을 고려하여 Camera 좌표계로의 좌표로 변환한다. 마지막으로 Camera의 내부 정보를 포함하는 K행렬을 통해 Image 상에서의 좌표 u , v 를 구하게 된다. 위의 K행렬은 간단한 편홀 카메라를 기준으로 작성되었다. 두 행렬의 곱을 H 로 하여 Homography Matrix로 정의한다. s 는 카메라 좌표계에서의 z 좌표로 최종 결과의 비례 상수로 나타난다. Fig 7는 위 관계식의 유도 과정과 좌표계의 관계를 도식화한 것이다.

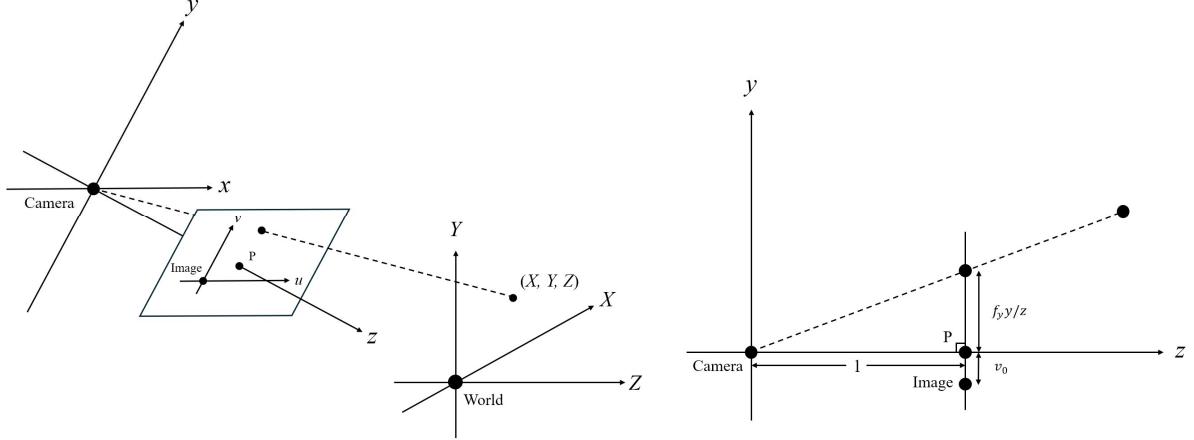


Fig 7. 좌표계 사이의 관계(좌)와 원근 사영 계산 과정(우)

하지만 IPM에서는 원근 사영의 역으로 Camera 좌표계에서의 이미지를 BEV 데이터로 변환해야 한다. 이를 위해 Backward 방식으로 IPM을 적용한다. Backward 방식은 BEV 데이터의 각 픽셀에서 원본 이미지의 픽셀로 접근하여 값을 가져오는 방법이다. 이는 BEV 이미지의 모든 픽셀이 채워지도록 보장된다. Fig 8는 자율 주행 시스템에서 IPM을 3차원 공간에 World, Camera 좌표계와 Image로 표현한 것이다.

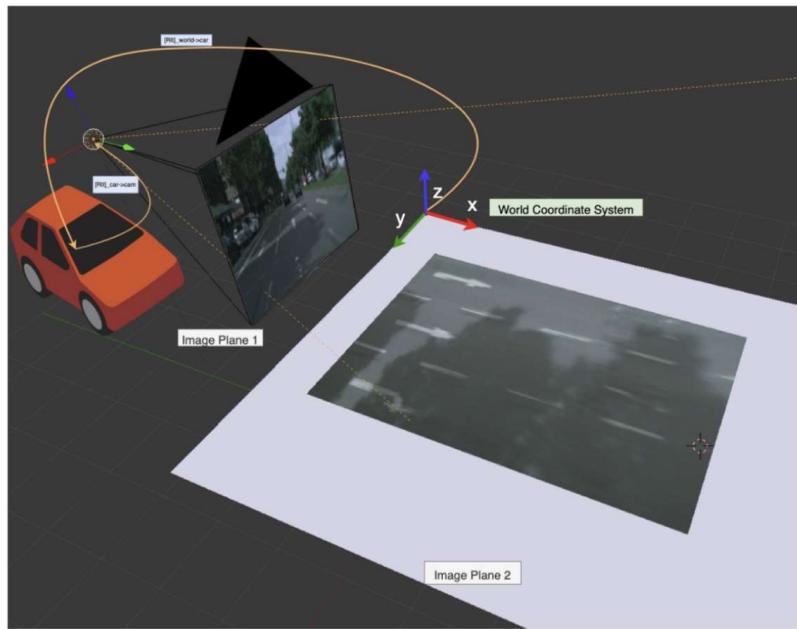


Fig 8. IPM에 사용되는 좌표계와 이미지의 관계도[17]

IPM은 World 좌표계의 3차원 공간 데이터를 2D 이미지 데이터로부터 얻어낸다. 차원이 감소하며 발생한 정보의 손실이 존재하는 이미지를 역 변환하는 것이기 때문에 물체의 Z 를 가정하여 발생하는 정보 손실을 해결한다. 자율 주행 시스템에서는 도로의 높이인 0을 Z 를 사용하나, 본 연구에서 활용하는 Crowd Localization은 개별 개체의 머리를 검출하기에 평지라고 보기에

는 어려움이 있는 상황이다. 그래서 본 연구에서는 평균 키인 1.7 m를 고정된 Z로 사용하여 IPM을 적용한다.

수식 (4)는 카메라의 보정 데이터가 주어진 상황에서 직접 Homography Matrix H를 계산할 수 있게 주어진다. 본 연구에서 활용하는 데이터는 카메라의 보정 데이터가 함께 주어지지 않는다. 하지만 차율 주행 시스템과 달리 대표적으로 사용되는 군중 데이터인 CCTV 영상 등은 지면이 고정되고 군중의 개별 개체가 움직이는 형태로 주어진다. 즉, 이미지에 나타나는 공간적인 정보를 이용하면 카메라 보정 데이터 없이 역으로 H를 추정할 수 있다. 이를 가능하게 하는 것이 Direct Linear Transformation(DLT)이다[18].

DLT는 입력으로 World 좌표계의 (X, Y, Z) 와 대응되는 Image에서의 (u, v) 을 쌍으로 여러 개 입력받는다. 이를 (X_i, Y_i, Z_i) , (u_i, v_i) ($i = 1, 2, 3, \dots, n$)이라 하자. 수식 (4)를 정리하면 각 쌍에 대해 수식 (5)의 관계식을 얻을 수 있다. 이때, h_{jk} 는 Homography Matrix H의 성분을 의미한다.

$$\begin{cases} -X_i h_{11} - Y_i h_{12} - Z_i h_{13} - h_{14} + u_i(X_i h_{31} + Y_i h_{32} + Z_i h_{33} + h_{34}) = 0 \\ -X_i h_{21} - Y_i h_{22} - Z_i h_{23} - h_{24} + v_i(X_i h_{31} + Y_i h_{32} + Z_i h_{33} + h_{34}) = 0 \end{cases} \quad (5)$$

수식 (5)의 결과를 모든 쌍에 대해 나열하여 수식 (6)과 같은 행렬 방정식의 형태로 작성할 수 있다.

$$Ah = \begin{bmatrix} -X_1 - Y_1 - Z_1 - 1 & 0 & 0 & 0 & u_1 X_1 & u_1 Y_1 & u_1 Z_1 & u_1 \\ 0 & 0 & 0 & 0 & -X_1 - Y_1 - Z_1 - 1 & v_1 X_1 & v_1 Y_1 & v_1 Z_1 & v_1 \\ \vdots & \vdots \\ -X_n - Y_n - Z_n - 1 & 0 & 0 & 0 & u_n X_n & u_n Y_n & u_n Z_n & u_n \\ 0 & 0 & 0 & 0 & -X_1 - Y_1 - Z_1 - 1 & v_n X_n & v_n Y_n & v_n Z_n & v_n \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{31} \\ h_{32} \\ h_{33} \\ h_{34} \end{bmatrix} = 0 \quad (6)$$

수식 (6)에서 구성된 행렬 A에 Singular Value Decomposition(SVD)를 적용하면, 최소 자승 해법인 h를 구할 수 있고, 이를 통해서 Homography Matrix H를 구할 수 있다. H의 독립 성분이 11개이고, 매 쌍마다 2개의 선형 방정식을 제공하므로 최소 6개의 독립 대응쌍이 주어지면 H를 구성할 수 있다.

6) 칼만 필터(Kalman Filter)

칼만 필터(Kalman Filter)는 시스템에서 상태 변수를 추정하기 위한 재귀 알고리즘으로, 노이즈가 존재하는 측정값을 이용하여 시스템의 현재 상태를 예측하고 갱신하는 과정을 반복한다. 이 필터는 제어 이론, 신호 처리, 항법 시스템 및 로보틱스와 같은 분야에서 주목할 만한 성과를 거두었다[19]. 본 연구에서 칼만 필터는 BEV 형태로 변환된 노이즈가 있는 현실 군중 데이터를 보정하기 위해서 Smoothing 기법으로써 도입되었다.

이번 연구에서 칼만 필터는 기본적으로 아래의 수식 (7)과 같은 선형 이산 시간 시스템 모델을 기반으로 한다.

$$\mathbf{x}_k = \mathbf{A}_{k-1} \mathbf{x}_{k-1} + \mathbf{w}_{k-1} \quad (7)$$

여기서 \mathbf{x}_k 는 k번째 시점에서의 군중 개체의 위치 정보를 담고 있는 상태 벡터, \mathbf{A}_{k-1} 은 상태 전이 행렬, \mathbf{w}_{k-1} 은 프로세스 노이즈를 의미한다. 현실 군중 데이터의 결과는 측정 결과 \mathbf{z}_k 로 수식 (8)과 같이 측정 노이즈 \mathbf{v}_k 를 포함하고 있다. 즉, 칼만 필터는 이 노이즈가 없는 값을 얻어내기 위한 확률적인 추론 문제를 풀어 현실 군중 데이터의 소음을 제거하는 것이다.

$$\mathbf{z}_k = \mathbf{x}_k + \mathbf{v}_k \quad (8)$$

칼만 필터는 예측 단계와 갱신 단계로 구성된다. 예측 단계에서는 이전 상태를 바탕으로 현재 상태를 예측하며, 갱신 단계에서는 실제 측정값을 이용하여 예측된 상태를 수정한다. 이 갱

신 단계에서 노이즈의 분포를 조정하게 되는 것이다. 결과적으로 매 단계에서 예측된 상태를 이용하면 Smoothing이 진행된 데이터를 얻어낼 수 있게 된다.

7) Dynamic Time Warping(DTW)

Dynamic Time Warping(DTW)는 서로 다른 두 시계열 간의 유사성을 평가하는 알고리즘으로, 누적 거리가 최소화되도록 매칭 경로를 찾아낸다. 두 시계열의 값을 동일한 시간대 기준으로 유clidean 거리를 사용해 비교하면, 전체 패턴이 유사하더라도 시간 차이에 따른 변화를 반영하지 못해 데이터의 미세한 차이나 떨림으로 인해 유사도 값이 크게 달라질 수 있다. 반면, DTW는 동일 시간대의 데이터뿐만 아니라 다른 시간대의 데이터도 비교하여 더 유사한 요소끼리 매칭함으로써 이러한 문제를 해결한다. Fig 9는 유clidean 거리 방식과 DTW를 적용하여 시계열의 각 점을 매칭시키는 방식을 시각적으로 나타낸 것이다. 그림에서 볼 수 있듯이, DTW는 어긋난 시계열에서도 효율적으로 일치하는 패턴을 찾아낸다. 본 연구에서는 시간에 따른 현실 군중 데이터 상에서의 군중 움직임과 시뮬레이션에서의 군중 움직임을 비교하여 전략의 현실성을 평가하는 과정이 포함된다. 이를 위해서 움직임의 패턴 유사도를 명확히 반영 가능한 DTW를 적용했다.

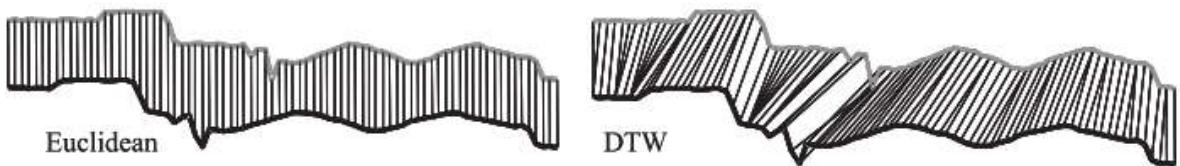


Fig 9. Euclidean Distance, DTW를 사용한 시계열 분석[20]

일차원 시계열 R_i, Q_i 사이의 최적 매칭은 수식 (9)를 적용해서 계산된다[20]. 식의 $\psi(k, j)$ 은 시계열 R_i, Q_i 의 (k th, j th) 원소까지의 최소 누적 거리이다. r_i^k, q_i^j 는 R_i, Q_i 시계열의 k 번째, j 번째 원소를 뜻한다.

이번 연구에서는 현실과 시뮬레이션 둘에서 얻어낸 2차원 상에서 군중의 위치를 데이터로 활용한다. 즉, 다차원 시계열(Multi-Dimensional Time Series, MDT)을 이용하는 것이다. 이 경우 각 차원이 독립적인 DTW_I 와 종속적인 DTW_D 의 2가지 유사도 측정 방법을 사용할 수 있다. DTW_I 는 M차원의 시계열 R, Q 의 차원을 각각 독립적으로 분석하여 수식 (10)과 같이 합을 $DTW(R, Q)$ 로 정의하여 사용하는 방법을 이용한다. DTW_D 에서는 모든 차원의 의존성을 가정하여 r^k, q^j 가 R, Q 의 k 번째, j 번째 원소일 때, 거리를 수식 (11)처럼 정의하고 일차원 시계열과 같은 알고리즘을 사용하게 된다. 현실 군중의 위치 데이터는 2차원의 좌표 형태로 주어지고, 이 둘은 종속적인 관계임이 명백하다. 따라서 본 연구에서는 DTW_D 를 적용하여 움직임 유사도의 척도로 사용하였다[21].

8) 베이지안 최적화(Bayesian Optimization)

본 연구 단계에서 전략을 구현할 때, 전략에 많은 조정가능한 파라미터를 포함하고 있다. 전략을 현실 군중 데이터와 비교하며 최적화하는 과정에선 이를 파라미터를 변경하며 DTW 거리가 최소가 되는 탐색 과정이 필요하다. 베이시안 최적화(Bayesian Optimization)는 비선형, 비용이 큰 함수의 최적화를 위해 널리 사용되는 방법론이다. 이 방법은 확률적 접근을 통해 함수의 불확실성 모델링하여 효율적으로 탐색하는 것이 특징이다. 베이시안 최적화의 핵심 아이디어는 가우시안 프로세스(Gaussian Process)를 활용해 목표 함수의 사전 확률 분포를 구축하고, 획득 함수(Acquisition Function)를 사용해 다음 점을 선택함으로써 최적화를 진행한다는 것이다[22].

전통적으로 사용되는 그리드 서치(Grid Search)와 랜덤 서치(Random Search)를 사용할 수도 있을 것이다. 그리드 서치는 사전에 정의된 파라미터 값들의 조합을 모두 탐색하는 방법으로, 탐색 공간이 작을 때는 유용할 수 있지만, 고차원 탐색 공간에서는 계산 비용이 급격히 증가하는 단점이 있다. 또한, 탐색 공간이 고정되어 있어 중요한 파라미터 값을 놓칠 가능성이 있다. 랜덤 서치는 사전 정의된 분포에 따라 임의의 파라미터 값을 샘플링하는 방법으로, 그리드 서치보다 효율적일 수 있으나, 여전히 고차원 공간에서의 탐색 효율성은 떨어질 수 있다.

이에 비해, 베이시안 최적화는 이전 평가 결과를 바탕으로 탐색을 수행하므로, 그리드 서치와 랜덤 서치에 비해 훨씬 적은 평가 횟수로 최적의 파라미터를 찾을 수 있다. 본 연구에서 최적화를 위해 사용되는 DTW 거리는 움직임 유사도 측정 척도로 시뮬레이션 전 과정을 돌려야 하는 매우 비용이 큰 함수이다. 즉, 베이시안 최적화를 사용한 계산 비용 절약이 필수적이다.

2. 연구방법 및 이론

가. 도구 및 재료

프레임워크 전반은 Python 3.8.19 기반의 가상환경 상에서 구현되었다. CPU 환경에서 테스트되었으며, GPU를 이용한 가속은 사용하지 않았다. 가상환경에서 활용한 주요 패키지와 모듈은 Table 1과 같다.

Table 1. 주요 패키지와 모듈

이름	버전	용도
numpy	1.23.5	행렬 연산
pandas	2.0.3	데이터 관리
matplotlib	3.7.5	데이터, 결과 시각화
opencv-python	4.10.0.84	영상 및 이미지 시각화
ffmpeg-python	0.2.0	영상 및 이미지 관리
imageio	2.34.2	GIF 관리
moviepy	1.0.3	영상 관리
fastdtw	0.3.4	DTW 적용
scipy	1.10.1	선형 대수 연산
torch	2.3.1	IIM 모델 적용
scikit-optimize	0.10.2	베이지안 최적화 적용

사용한 IIM 모델은 5109장의 군중 이미지를 포함하는 NWPU-Crowd 데이터셋을 통해서 학습되었으며, IIM 모델을 위한 라벨링은 선행 연구자에 의해 진행되었다. NWPU-Crowd 데이터는 이미지 데이터로 영상 데이터로부터 현실 군중 데이터를 변환하는 과정이 잘 이루어지는지 확인하기 위해 한국지능정보사회진흥원의 지원을 받아 구축된 ‘실내외 군중 특성 데이터’를 활용하였다. 이 데이터는 AI 허브(aihub.or.kr)에서 다운로드 받을 수 있다. 400개의 영상에서 얻어진 228,195장의 군중 이미지를 포함하고 있다.

전체 프레임워크를 테스트하기 위해서 분석 예시 시나리오를 설정하였다. 예시 시나리오의 장소는 도쿄, 시부야 스크램블 교차로의 횡단보도로 여러 방향으로 이동하는 군중이 혼합되어 복잡한 유동을 보이는 대표적인 사례이다. 이를 위해서 횡단 상황을 촬영한 영상 3개를 iStock에서 수집하였다. 수집한 영상의 스톡 비디오 ID는 1474927019, 1167282548, 1167927700이다.

나. 제작 과정

1) 전략 기반 CFD 시뮬레이션 제작

가) 물리 모델의 구현

CFD 모델은 군중이 이동하는 상황을 현실과 최대한 유사하게 묘사하는 것을 목적으로 DEM을 기반으로 구현했다. 모델의 요소로는 군중(Human)과 장애물(Obstacle)이 있다. 군중은 DEM에서의 이산 요소로 구현된다. 군중은 반지름(Radius), 질량(Mass), 속력(Speed) 등의 스칼라 속성과 위치(Position), 속도(Velocity), 가속도(Acceleration), 힘(Force), 목적 이동 방향(Direction) 등의 벡터 속성을 가진다. 기본적으로 모델은 각 프레임마다 LCM을 이용해 이웃 개체 쌍을 구한 이후, 전략에 의해 결정되는 심리적 상호작용과 물리적 법칙에 의해 결정되는 물리적 상호작용에 의한 힘이 계산된다. 이후 Verlet Integration 알고리즘을 통해 속도와 위치가 업데이트되는 방식으로 그 움직임이 구현되었다. 장애물은 현실에서의 벽 혹은 구조물의 역할을 한다. 장애물은 다각형 형태로, 군중에 비해 상대적으로 무한한 질량을 가졌다고 가정한다. 모델이 현실과 최적화할 수 있도록, 모델은 다양한 파라미터를 통해 구성되어 주어진 틀 속에서 자유도를 형성하였다. Fig 10는 DEM을 통해서 구현되는 군중과 장애물의 관계도를 보여준다.

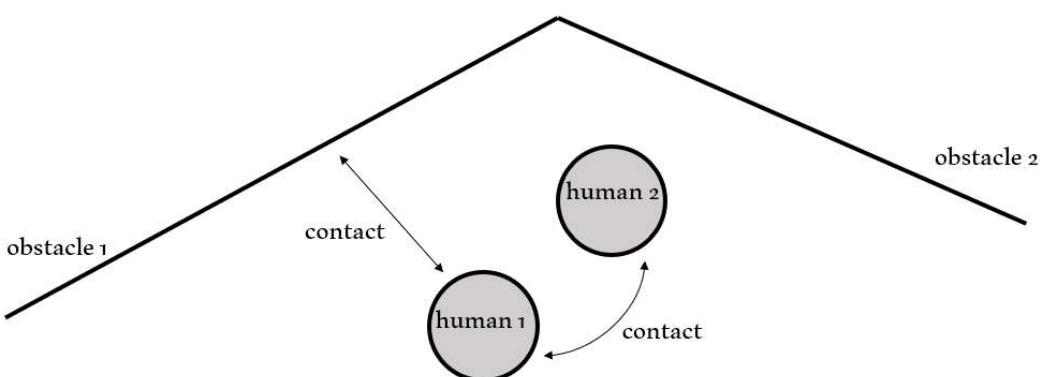


Fig 10. DEM을 통해서 구현된 군중과 장애물의 관계

구체적으로 본 시뮬레이션은 인접 개체 쌍 탐지, 심리적 상호작용의 계산, 물리적 상호작용의 계산, Verlet Integration의 과정을 반복한다. 심리적 상호작용과 물리적 상호작용의 계산을 위해서는 어떠한 군중 주변의 군중과 장애물 배치가 어떤 식으로 이루어져 있는지 구하는 것이 필연적이다. 하지만 모든 군중과 장애물 사이의 거리를 다 계산한다면 시간 복잡도 $O(N^2)$ 이 되어 처리 속도가 느리다. 따라 각 반복이 시작될 때 LCM 알고리즘을 통해 미리 모든 인접

한 요소를 시간 복잡도 $O(N)$ 으로 계산한다. LCM의 적용을 통해 어떠한 두 개체 사이의 거리가 두 개체의 크기보다 짧으면 두 개체를 물리적으로 접촉해 있는 상태로 간주하고, 어떠한 두 개체 사이의 거리가 두 개체의 크기의 χ 배보다 짧으면 두 물체의 위치가 인접한 상태인 이웃으로 간주한다. χ 는 모델의 파라미터이다. 모델에서 인간의 심리적 상호작용을 구현하는 방식은 일반적인 군중의 움직임 전략이 제시될 때 해당 전략의 결과에 기반해 군중이 이동하도록 설계되었다. 현재 군중의 상태 s 를 제공받으면 군중이 움직일 방향 a 를 반환하는 함수 $Q(s)$ 가 있다고 가정하자. 이 함수는 상태에 기반해 전략을 제시하는 전략 함수이다. 모델은 군중의 상태로 군중의 목적 이동 방향(Direction), 군중의 이웃 목록을 s 에 제공해 Q 의 반환 값 a 로 최종적으로 움직일 방향을 결정한다. 이를 통해 방향은 a , 세기는 군중의 보폭인 벡터로 군중이 목표하는 속도 벡터를 얻을 수 있다. 본 모델은 현재 군중의 속도 벡터와 군중의 목표 속도 벡터 사이 차이에 비례하는 군중의 의지에 의한 힘(구동력)을 추가해 심리적 상호작용을 구한다. Fig 11은 이런 심리적 상호작용의 구현 방식을 나타낸다. 이때의 비례계수 f_m 는 모델의 파라미터이다.

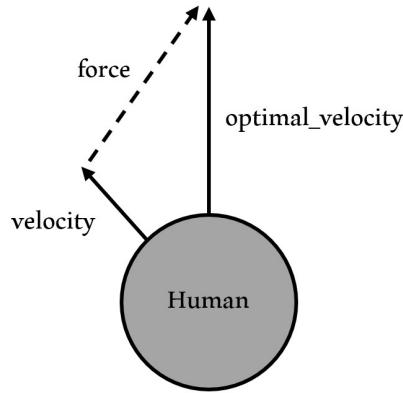


Fig 11. 군중 객체의 심리적 상호작용 구현 방식

물리적 상호작용의 계산은 군중과 군중, 혹은 군중과 장애물 사이의 2차원 비탄성충돌의 계산으로 이루어진다. 물리적 상호작용의 세기는 Soft Sphere DEM에서 흔히 사용하는 방식인 두 요소 사이 겹침 정도에 비례하도록 설계되었다. 이때의 비례계수 f_p , 충돌의 탄성계수 k 는 모델의 파라미터이다. 시뮬레이션의 매 반복의 마지막 부분은 Verlet Integration 알고리즘을 통해 군중의 위치와 속도를 업데이트하고 가속도와 힘을 초기화하는 과정이다.

나) 전략의 반영

시뮬레이션을 수행하기 위해서 인간의 움직임을 모식화한 타당한 군중의 움직임 전략, 즉 군중과 군중 사이, 군중과 장애물 사이 심리적인 상호작용을 다루는 방법을 제시하였다. 인간은 자신이 전진하고자 하는 방향(시야 방향)에 존재하는 다른 인간, 혹은 벽을 장애물로 취급한다. 이때 주변에 장애물이 없다면 목표하는 방향으로 전진한다. 만약 주변에 장애물이 존재한다면, 인간은 장애물을 피해 가려는 성질을 가진다. 인간이 장애물을 피하는 것은 돌아가는 것, 즉 목표 방향의 회전으로 이루어진다. 이때 장애물이 가까이 있고, 장애물의 방향과 시야 방향 사이 각도가 작을수록 장애물이 전진에 더 방해되므로 목표 방향을 더 많이 회전시킨다. 군중의 전략 함수는 위와 같은 인간의 움직임을 구현하는 방식을 기반으로 설계되었다. 전략 함수의 매개 변수는 군중의 목표 이동 방향, 군중의 이웃 목록이다.

전략 함수는 다음 세 단계에 걸쳐 이동 방향을 결정짓는다.

- 1) 인간 주변의 장애물이 전진에 방해를 주는 정도 계산
- 2) 모든 장애물의 방해 정도 총합
- 3) 방해 정도에 따라 목표 위치로 가기 위한 이동 방향 결정

각각의 군중의 이웃은 군중의 목표 이동 방향에 영향을 준다. 군중에서 d 만큼 떨어지고, 목표 이동 방향과 각도 차이가 θ 인 군중의 이웃이 있다고 하자. 이때 군중이 회전하는 정도 r 은 θ , d 와 음의 상관관계를 가진다. 특히, θ 의 절댓값이 $\pi/2$ 를 넘어설 경우, 군중이 전진하는 데에 아무런 영향을 미치지 않아 고려할 필요가 없다. 따라 전략 함수는 θ 의 절댓값이 $\pi/2$ 이하인 이웃에 대해서만 계산된다. d 는 군중의 반지름에 의해 그 최솟값이 제한되므로 r 과 d^{-2} 가 비례하는 관계를 가지도록 하였다. 하지만 θ 는 최솟값이 0까지 가능하므로, 단순히 나누어 줄 수 없다. 따라서 전략 함수는 θ 와 음의 상관관계를 가지고 0에서 극값을 가지는 아래의 함수를 이용해 r 을 계산한다. 이때 함수의 ‘날카로운 정도’를 나타내는 α 는 모델의 파라미터가 된다.

$$r = f(d, \theta) = \frac{\text{sign}(\theta)}{\alpha|\theta| + 1} \cdot d^{-2} \dots \quad (12)$$

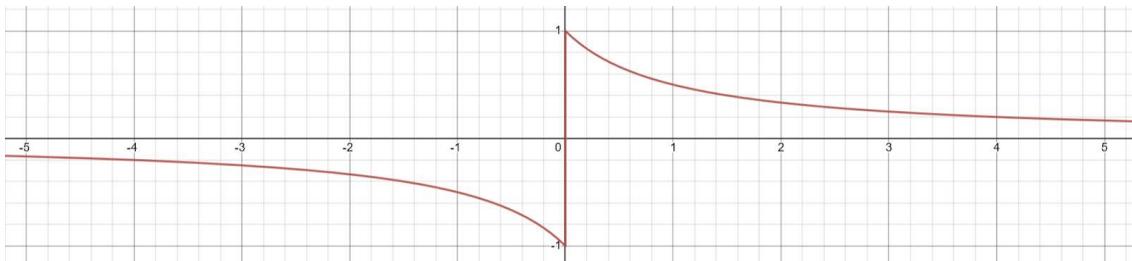


Fig 12. 수식 (12)의 그래프 개형

r 과 θ 사이의 비례함수는 Fig 12와 같은 개형을 띤다. θ 의 부호에 따라 r 의 부호가 변해 왼쪽으로 회전할지, 오른쪽으로 회전할지 결정된다. 군중의 이웃이 (d_i, θ_i) ($i = 1, 2, 3, \dots, N$)과 같은 목록으로 주어진다면 수식 (13)을 통해 군중의 최종적인 r -value를 구할 수 있다.

$$r = \sum_{i=1}^N f(d_i, \theta_i) \dots \quad (13)$$

이때 r 의 범위는 $(-\infty, \infty)$ 이지만 실제 군중이 회전할 수 있는 각도 범위는 정해져 있다. 군중이 회전할 수 있는 각도 범위를 $(-\pi/2, \pi/2)$ 로 가정하면 $(-\infty, \infty)$ 를 정의역으로 하고 $(-\pi/2, \pi/2)$ 를 치역으로 하고, 원점을 지나고, 단조 증가하는 조건을 만족하는 일종의 활성화 함수를 통해 최종적으로 회전할 각도를 결정하여야 한다. 이 활성화 함수는 분수함수를 응용하여 수식 (14)와 같이 만들었다. 이때 비례계수 β 는 모델의 파라미터이다.

$$\text{activation}(r) = \text{sign}(r) \cdot \left(\frac{\pi}{2} - \frac{1}{\beta|r| + 2/\pi} \right) \dots \quad (14)$$

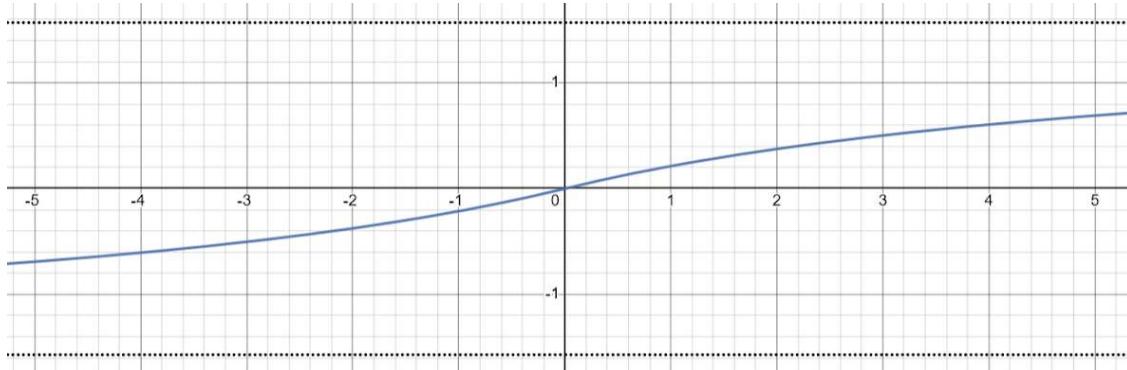


Fig 13. 수식 (14)의 그래프 개형

해당 함수는 Fig 13와 같은 개형을 띤다. 최종적으로 군중의 전략 함수는 군중의 초기 목표 이동 방향을 activation(r)만큼 회전한 방향으로 군중의 현재 목표 방향을 결과로 제시한다.

다) 다양한 상황에서 시뮬레이션 테스트

시뮬레이션이 의도한 것과 같이 군중의 움직임을 잘 묘사하는지 확인하기 위하여 다양한 상황을 설정하여 시뮬레이션의 결과를 확인하였다. 가장 기본적인 일자형 도로 상황, 군중 난류가 발생할 가능성이 높은 도로가 중간에 좁아지는 일자형 도로 상황, 다양한 방향으로 이동하는 군중이 교차하는 사거리와 팔거리 상황을 설정하여 시뮬레이션의 결과를 확인하였다.

2) 현실 군중 데이터의 변환

가) IIM 모델의 적용과 Crowd Localization

현실 군중 데이터를 BEV로 변환하기 위한 첫 단계로 IIM 모델을 적용하여 군중 이미지 데이터로부터 군중의 개별 개체의 위치에 대한 정보를 얻어내야 한다. 본 연구에서는 영상 데이터를 활용하기에 영상의 각 Frame마다 모델을 적용한다. 얻어진 IIM으로부터 군중의 수와 모든 점의 위치를 수치 데이터로 변환한다. Fig 14는 이 과정을 모식화한것이며, 얻어낸 수치 데이터의 형식 또한 나타낸다. 여기서 유의해야 할 점은 위치 데이터는 단순한 점 집합으로 나열되었다는 것이다. 개별 개체를 따로 분리하기 위한 정보는 IIM 모델을 통해선 얻을 수 없다.

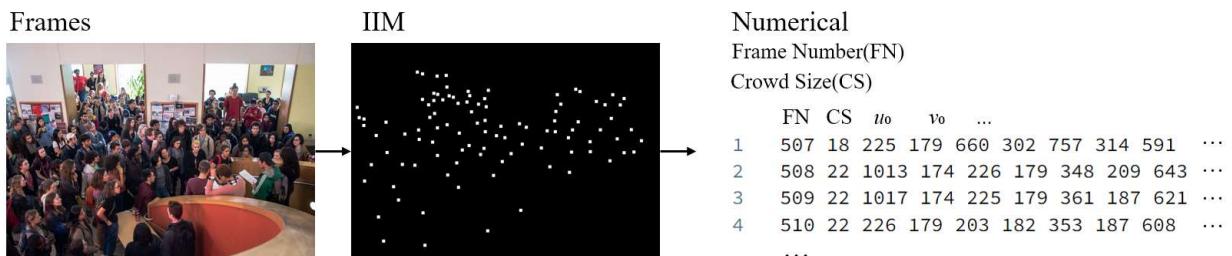


Fig 14. IIM 모델 적용 과정과 수치 데이터 형식

나) Homography Matrix H의 계산과 IPM 적용

DLT를 적용하여 IPM에서 요구되는 Homography Matrix H를 계산한다. DLT에 적용하기 위한 대응점 쌍은 직접 이미지로부터 수집하였고, 실제 측도를 바탕으로 World 좌표계에서의 값을 대응시켰다. 각 데이터셋에 대하여 10개 정도의 대응점 쌍을 이용하여 H를 계산하였으며, 완전한 편홀 카메라라는 가정을 적용하여 렌즈 왜곡의 효과는 무시하였다. 이 과정에서 독립성을 보장하기 위해서 $z \neq 0$ 이 아닌 데이터를 반드시 포함하였다. 계산된 Homography Matrix H를

가)의 Frame별 군중 위치 데이터에 적용하여 BEV 데이터로 변환하였다. 각각의 BEV 데이터는 개별 개체 집합의 World 좌표계에서의 위치 데이터를 포함하고 있다.

다) 개별 개체 분리 및 칼만 필터 적용

나)에서 구한 BEV 데이터의 개별 개체 집합 위치 데이터가 올바르게 개별 개체의 데이터로서 분리되기 위해선 얻어낸 점 집합들에 대한 개체 추적을 진행해야 한다. 할당 문제는 두 집합 간의 최적 매칭을 찾는 문제이다. 이 알고리즘에서 연속된 프레임에서 군중 데이터를 연결하는데, 이는 할당 문제와 유사하다고 볼 수 있다. 이전 프레임의 사람 위치와 현재 프레임의 사람 위치를 매칭하여 같은 사람을 추적하는 것이 목표이기 때문이다. 여기서 연결 비용을 의미하는 Cost는 점 사이의 Euclidean 거리로 주어진다. 할당 문제에는 전체 Cost 총합의 최소화를 목적으로 하는 Hungarian Algorithm 등이 사용된다. 하지만 본 연구는 전체 Cost의 총합을 최소화하는 것보다는 자연스러운 움직임의 연결이 목적이므로 최소 거리부터 매칭하는 Greedy Alrogithm을 적용했다. 알고리즘의 작동 방식은 순서대로 아래와 같다.

1. 초기화: 첫 번째 Frame의 각 점에 대해 새로운 트랙을 생성하고 고유 ID를 부여한다.
2. Cost Matrix 계산: 각 Frame마다 이전 Frame의 위치와 현재 Frame의 위치 간의 Euclidean 거리를 계산하여 Cost Matrix를 생성한다. Cost Matrix는 각 이전 위치와 현재 위치 사이의 거리를 나타내는 행렬이다.
3. 최소 Cost 매칭: Cost Matrix에서 가장 작은 값을 우선으로 하여 점들을 연결한다. 매칭된 점은 사용된 것으로 표시하고, 남은 점들을 계속 매칭한다. 만약 최소 비용이 설정된 임계 값을 초과하면 매칭을 중단한다.
4. 트랙 생성: 매칭된 점은 트랙에 추가하고, 매칭되지 않은 경계 근처 점은 새로운 트랙으로 간주하여 새로운 사람으로 추적을 시작한다. 일정 프레임 동안 매칭되지 않은 트랙은 추적을 종료하고 완료된 트랙으로 옮긴다.

추가적으로 이 알고리즘은 False Negative와 False Positive를 처리하기 위해 Missing Frames를 관리한다. 즉, Missing Frames를 관리함으로써 중간 Frame에서 인식되지 않더라도 특정 기준까지 트랙 추적 종료를 보류하는 조건을 넣어 False Negative를 해결한다. Missing Frames에 해당하는 점들은 개체 추적 완료 후에 선형 보간법을 이용하여 채워졌다. 또한 트랙의 길이가 1인 경우, Missing Frames가 기준을 넘어서면 이를 False Positive로 간주하여 트랙에서 제거한다. Fig 15는 개체 추적 알고리즘을 도식화한 것이다.

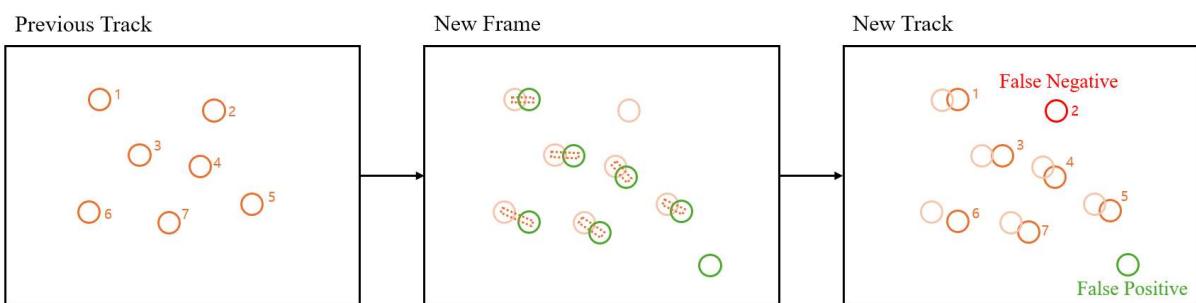


Fig 15. 개체 추적 알고리즘 도식화

이 알고리즘은 연속된 프레임에서 개체를 효과적으로 추적하며, 유실된 개체를 고려하여 실제 추적 환경에서 발생할 수 있는 오류를 최소화한다. 추적 완료 이후에 여러 과정에서 발생한 노이즈를 제거하고자 칼만 필터를 적용하여 트랙을 Smoothing했다.

라) 다양한 현실 군중 데이터로의 테스트

프레임워크에 가)에서 다)까지의 현실 군중 데이터 변환 과정이 적합하다는 사실을 확인하기 위해서 가).에서 소개한 ‘실내외 군중 특성 데이터’에 현실 군중 데이터 변환 과정을 테스트했다. 제공되는 데이터 중 DLT를 위한 실제 측도 바탕의 대응점 쌍의 수집이 가능한 Outdoor 촬영 영상을 사용했으며, 이에 해당하는 영상의 목록은 Table 2와 같다.

Table 2. 현실 군중 데이터 변환 테스트 영상 목록

장소	영상번호
신산공원	006, 007, 009, 010, 011, 012, 014, 015, 016
외도초등학교	110, 111, 112, 114, 086, 088
일도체육공원	117, 139, 140, 141, 142, 143, 145, 146, 148, 149, 150, 151, 153, 155, 156, 157, 158, 083
제주탑동광장	056, 063, 064

성능 평가는 나열한 영상 목록에 대한 시각화 결과를 통해서 정성적으로 이루어졌다. 추가적으로 Crowd Localization의 성능은 IIM 모델의 정밀도, 재현율 그리고 Frame당 연산 시간을 기록하여 정량적으로 이루어졌다.

3) 데이터 비교를 통한 전략 최적화 방안 제시

가) 움직임 유사도 측정 방법 고안

본 연구에서는 현실 군중 데이터와 시뮬레이션 데이터를 비교하여 전략의 최적화를 진행하는 것이 핵심적인 부분이다. 이를 위해 DTW를 적용한 움직임 유사도 측정 방안을 고안하였다. 선행연구 및 관련이론에서 언급했듯이 DTW는 시계열 데이터 간의 유사성을 측정하는 알고리즘으로, 두 시계열 데이터의 형태가 비슷하더라도 시간 축이 다를 경우 이를 정렬하여 유사도를 계산할 수 있는 강력한 도구이다. 본 연구에서는 현실 군중 데이터와 시뮬레이션 데이터를 각각 시계열 데이터로 변환한 후, DTW를 적용하여 두 데이터 간의 유사도를 측정하였다. 이 때, 전체 Frame을 구간별로 분할하고 각각의 초기 조건하에서 시간 변화를 비교하는 방식을 사용했다. 이를 통해 현실과 시뮬레이션 간의 차이를 정량화할 수 있었다.

나) 전략의 파라미터 최적화

DTW를 통해 측정한 유사도 결과를 바탕으로, 시뮬레이션 전략의 파라미터를 최적화하였다. 이 과정에서 베이지안 최적화를 적용하였다. 전략의 다양한 파라미터를 효율적으로 변화시키면서 DTW 거리가 최소가 되는 지점을 찾는 과정을 반복하였다. 이러한 최적화 과정은 시뮬레이션의 현실성을 높이고, 실제 군중의 움직임을 보다 정확하게 재현할 수 있도록 한다. 이를 통해 현실 군중 데이터를 기반으로 한 시뮬레이션의 타당성을 강화하고, 다양한 군중 상황에서의 대응 전략을 개선할 수 있는 기반을 마련하였다.

4) 예시 시나리오 적용: 도쿄, 시부야 스크램블 교차로의 횡단보도

프레임워크를 테스트해보기 위한 예시 시나리오로는 도쿄, 시부야 스크램블 교차로의 횡단보도를 선택하였다. 이 횡단보도는 여러 방향으로 이동하는 군중이 혼합되어 복잡한 유동을 보이는 대표적인 사례로, 본 연구의 방법을 적용하기에 적합한 환경을 제공한다. 이 시나리오에서는 각 군중의 이동 방향, 속도를 계산하여 시뮬레이션의 전략에 반영하였다. 그리고 현실 데이터와 전략을 통해서 계산된 시뮬레이션 속의 움직임을 비교했다. 이를 통해 복잡한 군중 유동 상황에서 연구에서 고안한 군중 유동 시뮬레이션 프레임워크의 효과를 검증하고, 현실 군중 데

이터를 기반으로 시뮬레이션의 타당성을 향상하는 과정을 확인했다.

본 연구에서 개발한 시뮬레이션 프레임워크를 시부야 횡단보도 시나리오에 적용하는 과정을 구체적으로 정리하면 다음과 같다.

1. **데이터 수집:** 시부야 횡단보도에서 촬영된 영상 형태로 현실 군중 데이터를 수집했다.
2. **IIM 모델 적용 및 Crowd Localization:** 수집된 데이터를 기반으로 IIM 모델을 적용하여 개별 개체의 위치를 식별하고, 이미지 상에서의 위치 정보를 데이터화했다.
3. **IPM 적용 및 개별 개체 분리:** 영상 이미지 상의 특정 지점들을 기준으로 Homography Matrix H를 계산하고 IPM을 적용했다. 이후 개체 분리 알고리즘을 통해서 현실 군중 데이터를 시뮬레이션 환경에 맞게 변환하였다.
4. **전략 기반 CFD 시뮬레이션 적용:** 변환된 데이터를 기반으로 각 개체의 이동 방향과 속도의 기본적인 정보를 얻어냈다. 이 정보를 활용해 시뮬레이션을 구성하여, 시뮬레이션에 현실과 대응되는 환경을 조성했다.
5. **DTW를 통한 데이터 비교 및 최적화:** Frame별로 구간을 나누어 각 구간의 초기 조건에 대한 시뮬레이션 결과와 현실 군중 데이터를 DTW를 통해 비교하고, 이를 지표로 하여 시뮬레이션 전략의 파라미터를 베이지안 최적화하여 유사도를 최대화하였다.

이와 같은 과정을 통해 도쿄, 시부야 스크램블 교차로의 횡단보도의 복잡한 군중 유동 상황을 효과적으로 시뮬레이션할 수 있었으며, 이는 본 연구의 시뮬레이션 프레임워크의 유효성을 입증하는 데 중요한 역할을 하였다. Fig 16은 시부야 스크램бл 교차로의 뷰비는 모습이다.



Fig 16. 도쿄, 시부야 스크램블 교차로

3. 결과

가. DEM을 적용한 전략 기반 시뮬레이션 제작

시뮬레이션이 의도한 것과 같이 군중의 움직임을 잘 묘사하는지 확인하기 위하여 다양한 상황을 설정하여 시뮬레이션의 결과를 확인하였다. Fig 17는 가장 기본적인 일자형 도로 상황의 시뮬레이션 결과이다. Fig 18는 군중 난류가 발생할 가능성이 높은 도로가 중간에 좁아지는 일자형 도로 상황의 시뮬레이션 결과이다. Fig 19는 다양한 방향으로 이동하는 군중이 교차하는 사거리 상황, Fig 20는 일방통행 팔거리 상황의 시뮬레이션 결과이다. 설정한 조건은 지난 이태원 참사 때 핵심적으로 군중 난류를 일으켰다고 평가받는 많은 인원, 다양한 진행 방향, 골목에 설치된 장애물 3가지 요인을 반영하여 설정되었다.

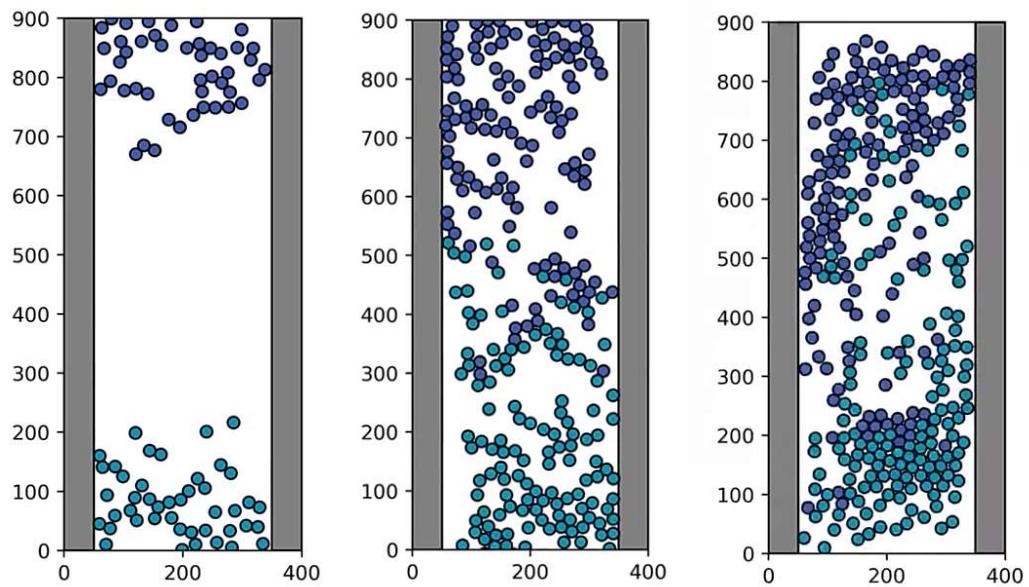


Fig 17. 기본적인 일자형 도로

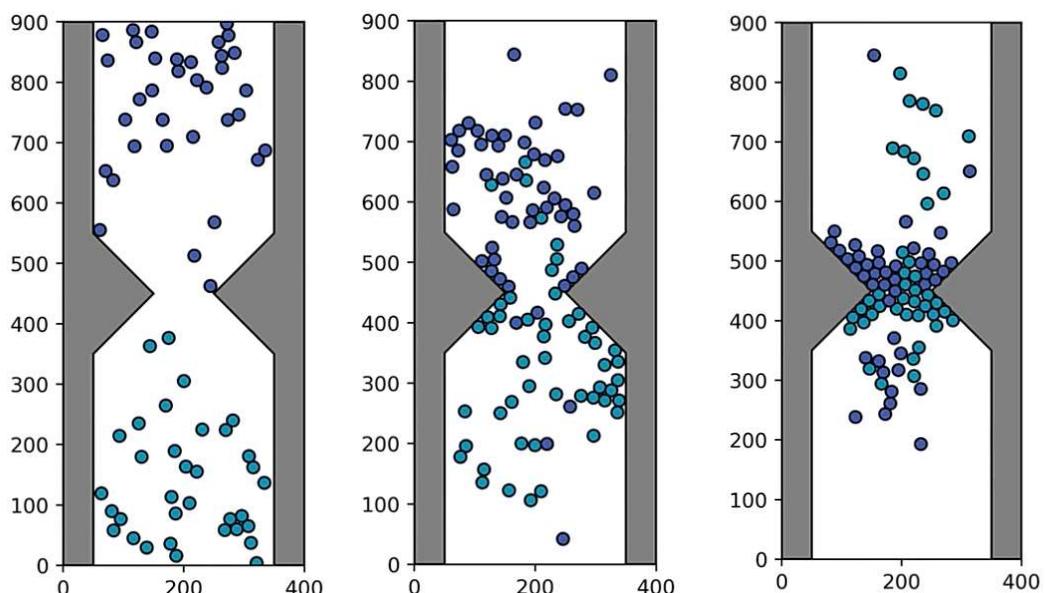


Fig 18. 좁아지는 일자형 도로

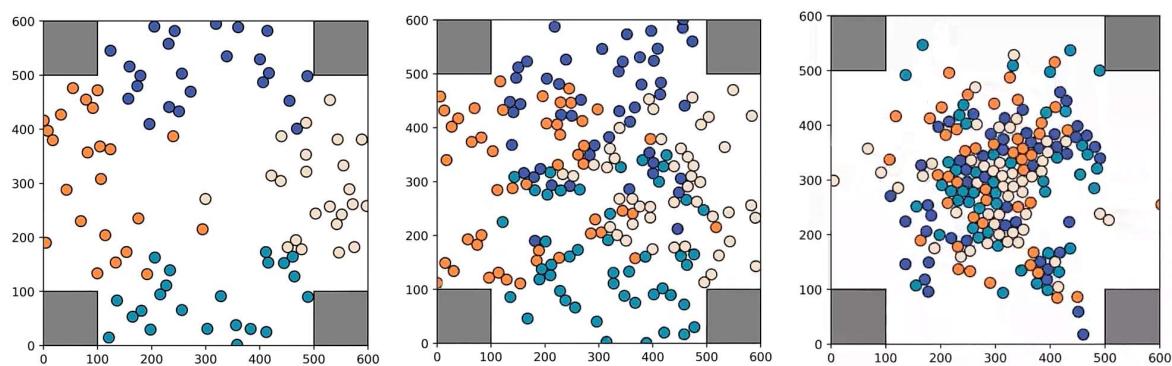


Fig 19. 사거리 상황

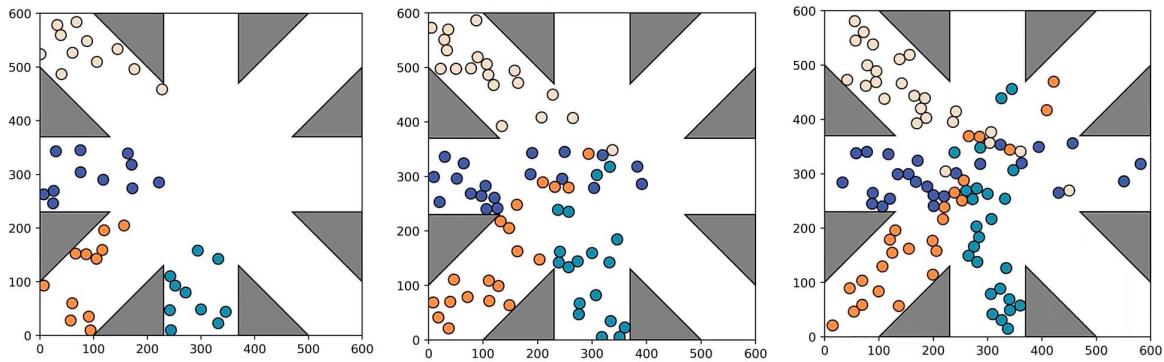


Fig 20. 팔거리 상황

나. 현실 군중 데이터의 변환

1) IIM 모델의 적용과 Crowd Localization

Table 3에 테스트 영상에 IIM 모델을 적용하여 구한 평균 Frame당 연산 시간, 평균 정밀도, 평균 재현율을 나타내었다. 모든 이미지는 720×1280 픽셀로 변환되어 연산되었다. 연산 시간은 Frame당 6에서 8초 정도로 나타났으며, 평균 정밀도는 0.8에서 0.9, 평균 재현율은 0.8정도로 나타났다.

Table 3. 장소별 IIM 모델의 성능 측정 결과

장소	평균 연산 시간(s)	평균 정밀도	평균 재현율
신산공원	6.89	0.922	0.874
외도초등학교	7.33	0.863	0.827
일도체육공원	8.56	0.842	0.832
제주탑동광장	7.96	0.913	0.844

2) 다양한 현실 군중 데이터로의 테스트

다양한 데이터에 대해서 현실 군중 데이터 변환 과정을 테스트 했으며, 보고서에는 최악의 경우인 Worst Case와 이상적인 경우인 Best Case를 첨부하여 정성적으로 분석하였다. Worst Case는 고찰에서 설명할 현실 군중 데이터 변환의 문제점이 모두 드러났으며, Best Case의 경우에는 가장 이상적으로 작동하여 문제가 없었다.

가) Worst Case

‘일도체육공원143’ 시나리오 데이터를 사용한 시각화 결과이다. 아래의 Fig 21은 Frame 352에서의 이미지에 IIM 모델을 적용한 결과이다. Fig 22는 Frame 352에서의 IPM과 개체 분리를 적용한 후 최종 변환 결과를 원본 이미지에, Fig 23는 BEV 형태로 나타낸 결과이다. BEV 형태의 데이터 상에서 붉은 영역은 원본 이미지상에서 관측할 수 없는 영역을 의미한다. Fig 22의 붉은 테두리가 Fig 23의 BEV 데이터의 관심 영역에 대응되는 원본 이미지의 영역이다.



Fig 21. ‘일도체육공원143’ 데이터, IIM 모델 적용 결과



Fig 22. ‘일도체육공원143’ 데이터, 개별 개체 트래킹 결과

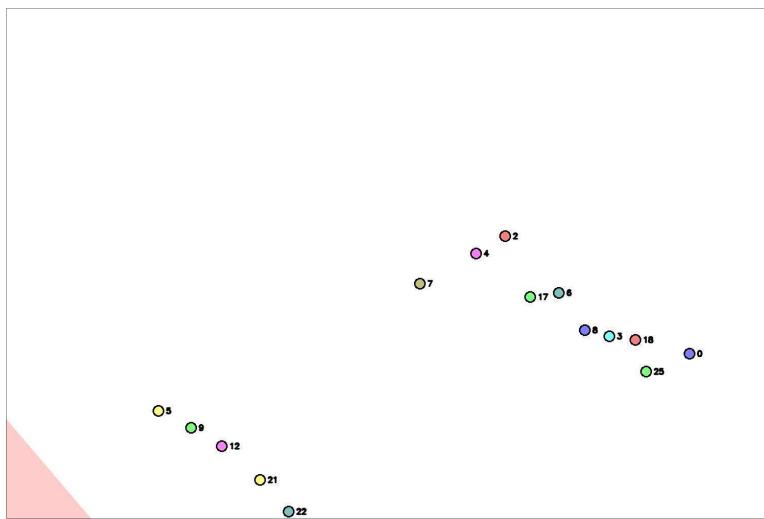


Fig 23. ‘일도체육공원143’ 데이터, BEV 데이터화 결과

나) Best Case

“제주탑동광장056” 시나리오 데이터를 사용한 시각화 결과이다. 아래의 Fig 24는 Frame 512에서의 이미지에 IIM 모델을 적용한 결과이다. Fig 25는 Frame 512에서의 IPM과 개체 분리를 적용한 후 최종 변환 결과를 원본 이미지에, Fig 26는 BEV 형태로 나타낸 결과이다. 위와 마찬가지로 해석의 편의를 위한 붉은 영역과 붉은 테두리를 나타내었다.

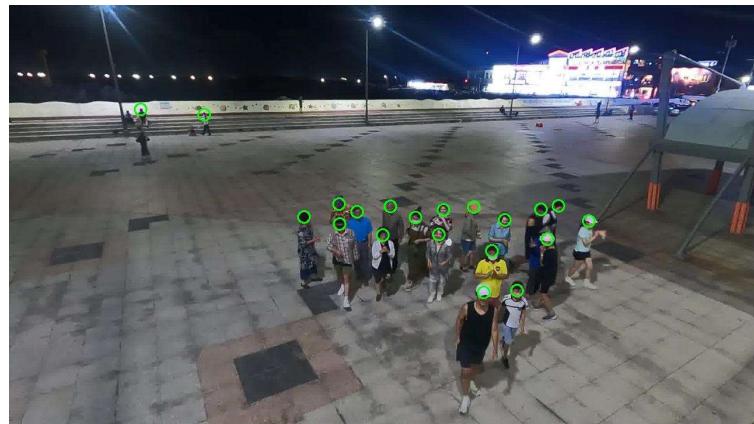


Fig 24. ‘제주탑동광장056’ 데이터, IIM 모델 적용 결과

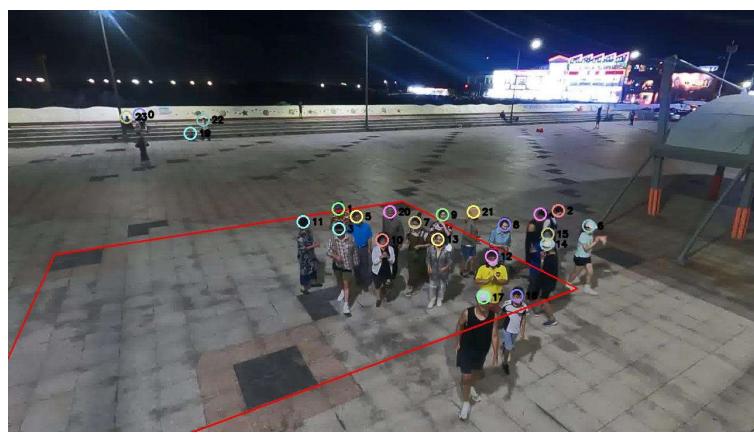


Fig 25. ‘제주탑동광장056’ 데이터, IIM 모델 적용 결과

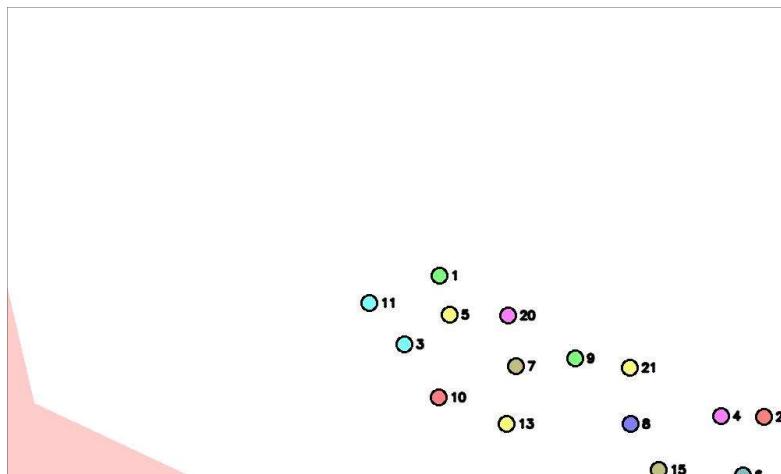


Fig 26. “제주탑동광장056” 데이터, BEV 데이터화 결과

다. 예시 시나리오 적용: 도쿄, 시부야 횡단 상황

1) 현실 군중 데이터 변환 결과

본격적으로 프레임워크를 테스트해보기 위해서 수집한 도쿄, 시부야 스크램블 교차로의 횡단 보도 영상에 현실 군중 데이터 변환을 시행한 결과가 Fig 27, 28, 29와 같다. 순서대로 IIM을 통한 Crowd Localization 결과, 개체 분리 결과, BEV 데이터화 결과이다.



Fig 27. 예시 시나리오 데이터, IIM 모델 적용 결과



Fig 28. 예시 시나리오 데이터, 개체 분리 결과

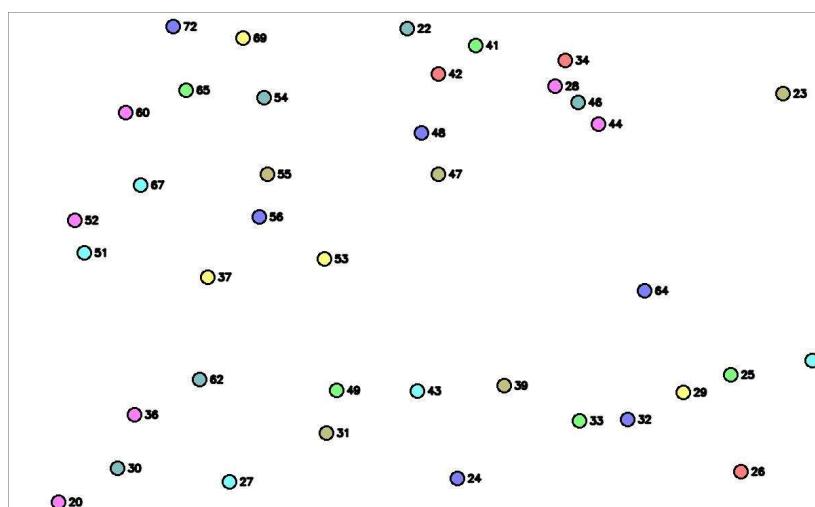


Fig 29. 예시 시나리오 데이터, BEV 데이터화 결과

2) 시뮬레이션 최적화

베이지안 최적화를 통하여 도쿄, 시부야 스크램블 교차로의 횡단보도 영상 현실 군중 데이터와, 동일한 초기 조건에서의 시뮬레이션 데이터 사이의 DTW 거리를 최소화하도록 모델의 파라미터를 최적화한 결과는 Table 4와 같다. 최종 DTW 거리는 약 36으로, 현실 군중의 이동과 시뮬레이션 군중의 이동은 평균 36 cm의 오차 내에서 같은 이동했다. Fig 30은 베이지안 최적화 결과 임의의 두 파라미터 쌍에 대한 모델 성능 변화 컬러맵을 나타냈다.

Table 4. 최적화된 모델 파라미터

이름	의미	값
α	전략 함수 뾰족한 정도	32520
β	전략 함수 비례계수	1
χ	이웃 판단 기준	4.10
f_m	심리적 상호작용 비례계수	985
f_p	물리적 상호작용 비례계수	124130
k	탄성 계수	0.7

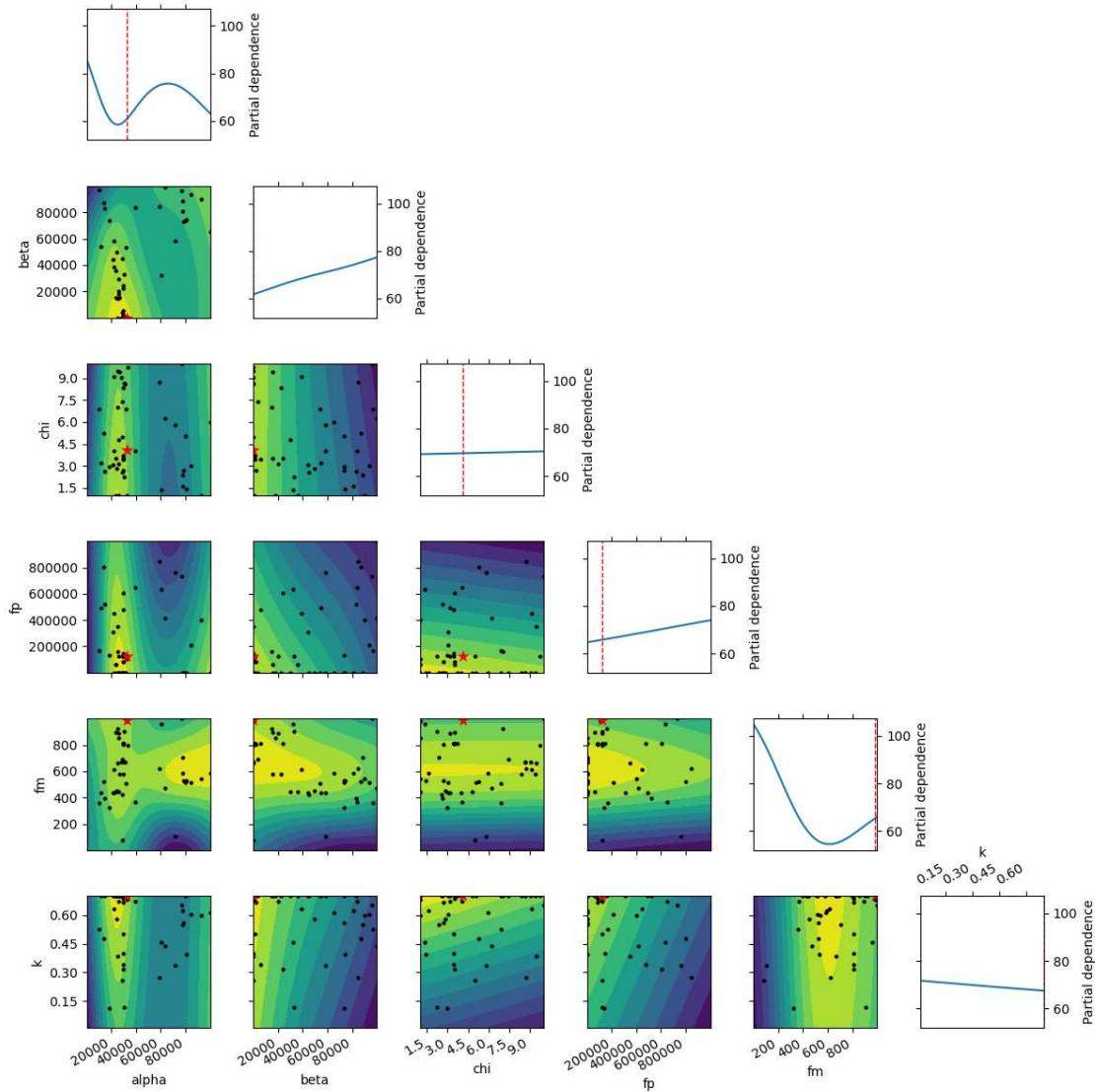


Fig 30. 임의의 파라미터 쌍 변화에 따른 모델 성능

4. 고찰 및 결론

이번 연구의 주요 목적은 CFD 시뮬레이션을 활용하여 현실 군중의 운동을 보다 정확하게 재현하고, 반영할 수 있는 새로운 CFD 시뮬레이션 프레임워크를 구현하는 것이었다. 이를 위해 크게 전략 기반 CFD 시뮬레이션 제작, 현실 군중 데이터의 변환, 데이터 비교를 통한 전략 최적화 방안 제시의 3단계를 프레임워크에 포함하고자 연구를 진행하였다.

연구 결과, 전략 기반 시뮬레이션은 다양한 시나리오에서 정성적, 정량적으로 군중의 운동 다운 결과를 도출하였다. 설정한 4가지의 테스트 시나리오 모두에서 직관적으로 군중의 운동과 가까운 모습을 보여주었으며, 사람 수, 장애물 설정 등을 자유롭게 진행할 수 있도록 제대로 제작되었음을 확인할 수 있었다. 하지만 앞서 선행연구에서 언급했듯이 직관적으로 인간다운 행동을 하는 것은 본 연구의 목적이 아니므로 전략을 좀 더 현실성 있게 최적화 하는 과정이 필요했다.

현실 군중 데이터 변환 과정 또한 테스트를 통해 대부분의 성공적인 변환이 이루어진다는 것을 확인할 수 있었다. 다만, 높이를 평균 키인 1.7 m로 고정한다는 점에서 나이, 성별 차이가 많거나 지형지물이 복잡한 경우에는 BEV 데이터 내에서 위치 데이터의 오차가 매우 커진다는 것을 확인할 수 있었다. 또한 군중이 매우 밀집하여 일부가 가려질 경우 Crowd Localization 단계에서 탐색되지 않는다는 본질적인 문제점이 있었다. 그러나, 이는 제언에서 제안할 후속 연구를 통해서 개선될 수 있으며, 많은 경우에 큰 지장을 주지 않는 문제점이라는 것을 확인할 수 있다.

최종적으로 전체 프레임워크의 성능을 검토하기 위해 예시 시나리오를 적용한 결과, 다양한 방향으로 움직이는 군중 유동 패턴을 효과적으로 시뮬레이션에 반영한다는 사실을 확인할 수 있었다. DTW와 베이지안 최적화를 통해서 복잡한 환경에서의 군중 유동을 정확하게 모델링하도록 최적화할 수 있었으며, 시뮬레이션 결과는 현실 데이터와 높은 유사도를 보였다. 이러한 결과는 새로운 프레임워크의 타당성을 입증하며, 군중 유동 역학 분야의 발전에 기여할 수 있는 중요한 기반이 되었다.

결론적으로, 본 연구를 통해 제시한 새로운 CFD 시뮬레이션 프레임워크는 군중 유동 문제를 해결하는 데 효과적임을 확인하였다. 예시 시나리오를 통해서 연구에서는 간단한 전략을 적용하였지만, 전략 부분은 유동적으로 변화시킬 수 있기에 좀 더 복잡한 전략 또한 적용 가능할 것이다. 이런 방식으로 접근을 확장한다면 스마트 시티 구축, 교통 관리, 안전 계획 등 다양한 분야에서 실질적인 활용 가능성을 제시할 수 있을 것이며, 군중 유동 분석의 정확성과 유연성을 크게 향상시킬 수 있다. 특히, 이태원 참사와 같은 실제 군중 밀집 사고의 예방 및 대처에 유용한 도구로 사용될 수 있을 것이다.

5. 제언

본 연구는 새로운 전략 기반 CFD 모델을 통해 군중 유동 문제를 효과적으로 분석하고 해결 할 수 있는 방법을 제시하였다. 그러나 이 연구에는 몇 가지 한계점이 있으며, 이를 보완하기 위한 향후 연구 방향을 제안한다.

첫째, 본 연구는 전략을 연구자들이 임의로 설정하고 몇 가지의 파라미터를 최적화하는 것에 그쳤다. 완전한 자유도가 보장되기 위해선 Deep RL을 적용한 연구에서와 같이 전략을 인공지

능 모델로 자유도가 높게 구현하는 것이 좋을 것이라고 판단된다. DTW 거리를 오차 함수로 학습한다면 좀 더 정교한 예측을 가능하게 하며, 각 개체가 동적으로 의사결정을 내리는 것 또한 구현할 수 있을 것이다. 본 연구에서는 컴퓨팅 리소스의 한계로 큰 연산량을 감당할 수 없어 이 부분을 제안만 할 뿐 실제로 실행하지 못했으나, 후속 지원을 통해서 이 부분을 해결할 수 있을 것으로 생각된다.

둘째, 다중 카메라를 이용한 현실 군중 데이터 변환화의 개선이 필요하다. 위에서 언급했듯이 공간 데이터의 손실로 인해 높이를 고정하는 과정, 개체끼리 서로 겹치는 문제 등이 데이터 변환의 오차를 발생시켰다. 이는 단일 카메라를 사용하기 때문에 발생하는 문제로 다양한 각도에서 수집된 다중 카메라 데이터를 통합한다면 군중의 움직임을 보다 정확하게 재현할 수 있을 것이다. 이를 통해, 개별 군중의 행동을 더욱 정밀하게 분석하고, 다양한 상황에서의 군중 유동을 보다 정확하게 시뮬레이션에 반영할 수 있을 것이다.

셋째, 다양한 인간 형태나 개인별 차이를 고려한 모델링을 진행할 필요가 있다. 이를 통해, 개인별 움직임의 차이나 다양한 인간 형태에 따른 군중 움직임의 복잡성을 더 잘 이해하고 예측할 수 있을 것이다.

본 연구는 새로운 전략 기반 CFD 모델을 통한 군중 유동 분석의 타당성 향상 가능성을 제시하였다. 향후 연구에서는 제시된 한계점을 극복하고, 더욱 개선된 모델을 개발하는 것이 중요하다. 이러한 개선된 모델은 군중 유동 분석의 새로운 표준이 될 것으로 기대하며, 다양한 분야에서의 실질적인 활용 가능성을 높일 수 있을 것이다.

6. 참고 문헌

- [1] Nam, S. (2019). CROWD-FLOW SIMULATION USING THE POTENTIAL PATHLINE METHOD. *WIT Transactions on Engineering Sciences*, 125, 27-35.
- [2] 남성원. (2020). 철도역에서 양 방향 군중 유동의 전산 해석. *대한기계학회 논문집 B 권*, 44(8), 475-482.
- [3] 남성원. (2020). 원활한 군중 유동을 위한 환승 보행로 개선에 관한 연구. *대한기계학회 춘추학술대회*, 720-721.
- [4] Langston, P. A., Masling, R., & Asmar, B. N. (2006). Crowd dynamics discrete element multi-circle model. *Safety Science*, 44(5), 395-417.
- [5] Xu, M. L., Jiang, H., Jin, X. G., & Deng, Z. (2014). Crowd simulation and its applications: Recent advances. *Journal of Computer Science and Technology*, 29(5), 799-811.
- [6] Sagredo-Olivenza, I., Cárdenas-Bonett, M., Gómez-Sanz, J. J., & Pavón, J. (2018, October). Using Graphs of Queues and Genetic Algorithms to Fast Approximate Crowd Simulations. In *Proceedings* (Vol. 2, No. 19, p. 1216). MDPI.
- [7] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.
- [8] Lee, J., Won, J., & Lee, J. (2018, November). Crowd simulation by deep reinforcement learning. In *Proceedings of the 11th ACM SIGGRAPH Conference on Motion, Interaction and Games* (pp. 1-7).

- [9] 손권중, 박준영. (2020). 이산요소법의 소개 및 응용 분야. *기계저널*, 60(9), 31-36.
- [10] O’ Sullivan, C. (2011). Particle-based discrete element modeling: geomechanics perspective. *International Journal of Geomechanics*, 11(6), 449-464.
- [11] Mattson, W., & Rice, B. M. (1999). Near-neighbor calculations using a modified cell-linked list method. *Computer Physics Communications*, 119(2-3), 135-148.
- [12] Verlet, L. (1967). Computer“ experiments“ on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules. *Physical review*, 159(1), 98.
- [13] Dummer, J. (2004). A simple time-corrected verlet integration method. *Game Developer*.
- [14] Swope, W. C., Andersen, H. C., Berens, P. H., & Wilson, K. R. (1982). A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *The Journal of chemical physics*, 76(1), 637-649.
- [15] Gao, J., Han, T., Wang, Q., Yuan, Y., & Li, X. (2020). Learning independent instance maps for crowd localization. *arXiv preprint arXiv:2012.04164*.
- [16] Khan, M. A., Menouar, H., & Hamila, R. (2023). Revisiting crowd counting: State-of-the-art, trends, and future perspectives. *Image and Vision Computing*, 129, 104597.
- [17] Ahmad, M. A., & Muad, A. M. (2021, November). Calibration of Inverse Perspective Mapping from Different Road Surface Images. In *2021 IEEE 11th International Conference on System Engineering and Technology (ICSET)* (pp. 139-142). IEEE.
- [18] Shapiro, R. (1978). Direct linear transformation method for three-dimensional cinematography. *Research Quarterly. American Alliance for Health, Physical Education and Recreation*, 49(2), 197-205.
- [19] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.
- [20] Keogh, E., & Ratanamahatana, C. A. (2005). Exact indexing of dynamic time warping. *Knowledge and information systems*, 7(3), 358-386.
- [21] Shokohi-Yekta, M., Hu, B., Jin, H., Wang, J., & Keogh, E. (2017). Generalizing DTW to the multi-dimensional case requires an adaptive approach. *Data mining and knowledge discovery*, 31(1), 1-31.
- [22] Wang, X., Jin, Y., Schmitt, S., & Olhofer, M. (2023). Recent advances in Bayesian optimization. *ACM Computing Surveys*, 55(13s), 1-36.