

모션 매칭을 활용한 상호작용적 캐릭터 경로 추적

이정민⁰, 권태수, 이윤상
한양대학교 컴퓨터소프트웨어학과
{j0064423, taesoo, yoonsanglee}@hanyang.ac.kr

Interactive Character Path-Following using Motion Matching

Jeongmin Lee⁰, Taesoo Kwon, Yoonsang Lee
Department of Computer Science, Hanyang University

요약

이 연구에서는 모션 매칭[1]과 입력 경로를 기반으로 사용자와 상호작용하며 캐릭터를 조종할 수 있는 방법을 제안한다. 사용자는 마우스 또는 태블릿 펜으로 입력한 경로에서 캐릭터의 속도도 함께 지정할 수 있으며, 이 과정에서 실제 사람의 이동 경로와 사용자의 입력 경로는 그 속도와 모양에서 차이가 날 수 있다. 이 연구에서는 이로 인한 오류를 최소화하는 방법들을 제안한다.

캐릭터가 매 시각 바라보는 방향은 간단히 정면만을 바라보도록 할 수도 있고, 이 연구에서 제안하는 RNN기반의 DirectionNet을 사용해서 예측할 수도 있다.

1. 서론

모션 매칭[1]은 게임 산업에서 캐릭터의 조작에 널리 쓰이고 있는 방법이다. 이 연구에서는 [1]에서와 마찬가지로 발의 위치와 속도에 특정된 특성 값을 사용했다. i 번째 프레임에서의 특성 값을 수식으로 표현하면 아래와 같다:

$$\text{feature}_i = \{p_i^{l\text{foot}}, v_i^{l\text{foot}}, p_i^{r\text{foot}}, v_i^{r\text{foot}}, v_i^{\text{hip}}, t_i, d_i\} \in \mathbb{R}^{27}$$

각 요소는 양쪽 발의 위치(p)와 속도(v), 캐릭터 루트의 속도(h), 그리고 캐릭터가 이 데이터베이스에서 3개 지점의 미래 프레임에서 가지는 루트의 2차원 위치(t)와 2차원 속도(d) 값을 포함한다. p , v , h 는 캐릭터 프레임의 현재 상태에 해당하며, t , d 는 미래의 정보로, 이 연구에서는 사용자가 입력한 경로로부터 t 와 d 를 새롭게 조합해서 캐릭터의 미래 동작을 원하는 대로 조작한다.

이 연구에서는 입력 경로에 속도 정보를 포함하여, 사용자가 경로를 그리는 속도에 따라 캐릭터가 다양한 속도로 부드러운 동작을 생성할 수 있다.

2. 경로 선 처리

사용자가 연속적으로 입력하더라도 컴퓨터에서는 이산적으로 저장하기 때문에 오차가 생길 수 있다. 따라서 이 연구에서는 입력된 각 점의 시간 간격을 함께 저장해서, 입력된 점들의 시간 간격이 150 Hz가 되도록 다시 샘플링 한 후, 각각의 점들이 30 Hz로 샘플링 된 것으로 간주하여 i 번째 점의 위치를 i 번째 프레임에서의 입력 경로 상의 위치 $p(i)$ 로 나타내었다.

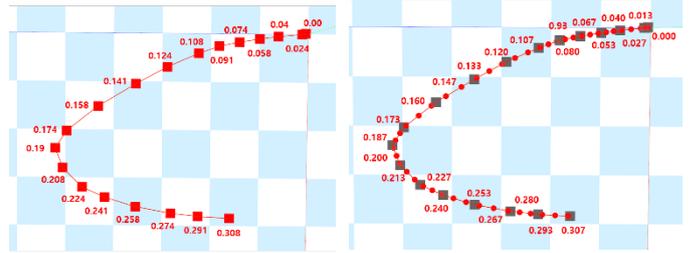


Figure 1: Input trajectory resampled in 150Hz.

또한 사용자가 입력한 점들은 실제 캐릭터의 움직임과 달리 마우스나 태블릿 펜의 가벼운 무게나 마찰 등으로 인해 더 과격할 수 있다. 예를 들어 급격한 방향 전환이 있을 때, 실제 사람은 훨씬 무겁고 발과 땅의 접촉으로만 속도를 바꿀 수 있기 때문에 속도 변화는 비교적 부드럽게 일어난다. 그러므로 이 연구에서는 입력된 경로의 간격을 가우시안 필터(크기 19, 표준 편차 3)를 활용하여 재조정했다.

3. 경로 위에서의 캐릭터의 목표 위치

경로 위의 점들로부터 모션 매칭 특성의 미래 경로 위치(t)를 계산하기 위해서는 우선 현재 캐릭터의 위치가 경로에서 어느 점에 해당하는지(p_d)를 알아야 한다. 이상적으로 캐릭터가 매 프레임 경로 위의 다음 점에 도달한다면 매번 다음 위치를 확인하기만 하면 되겠지만, 실제로는 경로가 빠르게 그려진 경우 경로 점들이 지나치게 빨리 소진되어 버린다. 또는 현재 캐릭터의 위치에서 가장 가까운 점으로 가정할 수도 있지만, 이런 경우 경로가 서로 얹힌 경우 무작위로 건너뛰어 버리거나, 무한히 같은 경로 조각을 쿼리하게 될 수도 있다.

이 연구에서는 p_d 를 계산하는 아래 식을 제안한다.

* 구두발표논문

* 본 논문은 요약논문(Extended Abstract)으로서, 본 논문의 원본 논문은 현재 타 학술대회(논문지)에 제출 중임.

* 본 연구는 정부(과학기술정보통신부)의 지원으로 한국연구재단의 지원을 받아 수행되었음(NRF-2019R1C1C1006778, NRF-2020R1A2C1012847).

$$i_d = \operatorname{argmin}_i \|p_c - p(i)\|$$

$$\text{s.t. } i_d^{\text{prev}} - \alpha \leq i \leq i_d^{\text{prev}} + \beta$$

이 연구에서는 이 식으로부터 현재 캐릭터의 위치 p_c 에서 제한된 탐색범위($\alpha=0, \beta=10$)내의 가장 가까운 점 $p(i)$ 를 찾아 경로 위의 i_d 번째 점 p_d 를 찾는다.

4. 쿼리의 미래 위치에 대한 속도 제한

입력된 경로로부터 속도가 변화하는 모션 매칭 쿼리(t)는 위에서 찾은 p_d 로부터 특성 데이터베이스를 만들 때와 같은 간격(10프레임)으로 점 3개를 샘플링해서 생성된다. 하지만 사용자가 그린 경로가 데이터베이스에 있는 동작들보다 훨씬 빠르게 그려졌다면 비현실적인 쿼리가 만들어지기 때문에 경로를 제대로 따라가지 못할 수 있다.

이 연구에서는 이런 경우, 제한된 속도로부터 가능한 최대 이동거리에 해당하는 경로 상의 점 $p(i_{\text{vmax}})$ 와 p_d 사이에서 시간축 상에서 삼등분된 위치의 점들을 t 로 사용한다.

5. 급격히 꺾인 경로에서 쿼리하기

사용자가 입력한 경로를 모션 매칭으로 따라가게 할 때, 입력된 미래 정보 쿼리 사이에는 시간 간격이 존재하기 때문에 캐릭터는 급격히 꺾인 경로를 따라가지 못할 수 있다. 이 연구에서는 만약 미래 위치 쿼리 t 에 모서리가 포함되는 경우(Fig. 2 (a)), 가장 안쪽(i_{corner})까지 도달하도록 하기 위해(Fig. 2 (b)) i_{corner} 직전의 미래 점 p_i^j 과 i_{corner} 를 이은 직선 상에 p_i^{j+1} 를 새로 정의하고(Fig. 2 (c)) 이후의 점들도 함께 평행이동시킨다(Fig. 2 (d)).

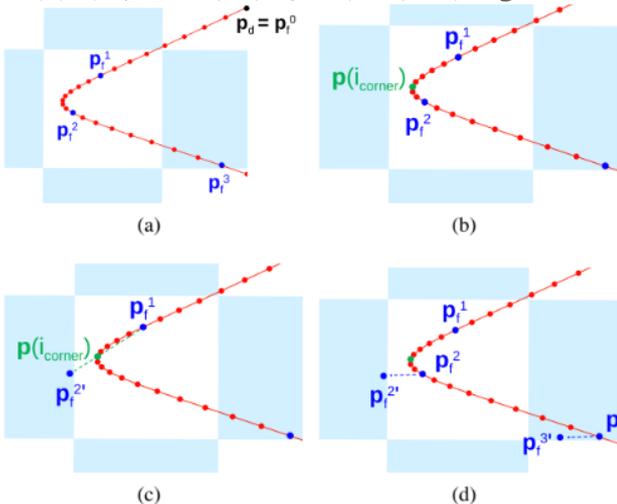


Figure 2: Modification of the future positions for sharp corners.

6. 미래 경로의 캐릭터 방향

모션 매칭 쿼리의 사용자 쿼리는 미래 위치 t 외에도 t 의

각 점에서 캐릭터가 바라보고 있는 방향 d 가 있다. 이 방향은 간단히 항상 앞을 향하도록(경로상의 접선 방향) 고정할 수도 있지만 데이터베이스에 있는 다양한 방향을 바라보는 동작을 고루 사용하기 위해서 간단한 GRU로 구성된 네트워크를 사용해 예측할 수도 있다. 이 연구에서는 DirectionNet을 제안한다.

DirectionNet은 Quaternet의 PaceNetwork[3]로부터 영감을 받았으며, 30개의 히든 유닛으로 이루어진 한 층의 GRU유닛과 또 한 층의 완전연결층으로 이루어져 있다. DirectionNet을 사용하면 과거의 캐릭터 위치와 바라본 방향을 바탕으로 쿼리로 입력된 미래 경로에서 바라볼 방향을 예측해, 옆이나 뒤를 향해 걷는 동작도 출력할 수 있다.

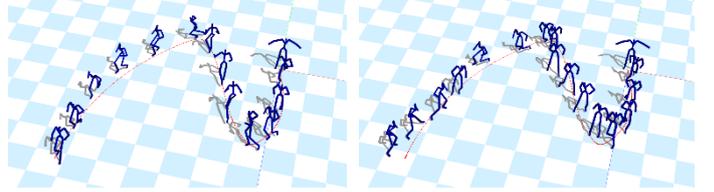


Figure 3: A comparison between tangential direction(left) and DirectionNet(right).

학습은 7만 프레임의 모션 데이터베이스와 1개의 GPU에서 10시간가량이 소요되었으며, 모션 데이터베이스는 4초 길이의 조각으로 나뉘어 2초의 과거 정보를 바탕으로 2초의 미래 방향을 예측하도록 했다.

7. 실험 결과 및 결론

이 연구에서는 여러 개의 캐릭터가 있거나 장애물이 산재하는 상황에서 캐릭터를 조종하는 실험으로, 2차원 경로를 입력으로 모션 매칭을 활용해 동작을 합성하는 방법의 효과를 보였다.

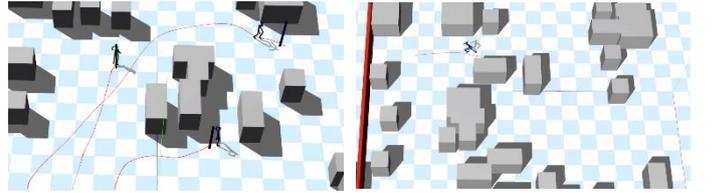


Figure 4: Interactive experiments.

또한 뾰족하게 꺾인 경로나 꼬인 경로 등 경로를 입력으로 처리하는 다른 연구들[2]보다 더 다양하고 복잡한 입력 경로에서도 부드러운 동작을 생성할 수 있음을 보였다.

참고문헌

- [1] S. Clavet, "Motion matching and the road to next-gen animation", *GDC 2016*
- [2] Daniel Holden, Oussama Kanoun, Maksym Perepichka, and Tiberiu Popa, "Learned motion matching", *ACM Trans. Graph. Vol. 39 Issue 4*, 2020
- [3] D. Pavllo, D. Grangier, and M. Auli, "Quaternet: A quaternion-based recurrent model for human motion," *British Machine Vision Conference (BMVC)*, 2018