



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2013-0133778
(43) 공개일자 2013년12월09일

- (51) 국제특허분류(Int. Cl.)
G06T 13/40 (2011.01)
- (21) 출원번호 10-2013-7013299
(22) 출원일자(국제) 2011년11월28일
심사청구일자 없음
(85) 번역문제출일자 2013년05월24일
(86) 국제출원번호 PCT/CN2011/083077
(87) 국제공개번호 WO 2012/069022
국제공개일자 2012년05월31일
(30) 우선권주장
PCT/CN2010/001899 2010년11월26일 중국(CN)

- (71) 출원인
톰슨 라이센싱
프랑스 92130 이씨레몰리노 찬 다르크 류 1-5
(72) 발명자
시아, 지진
중국 100192 베이징 하이 디안 디스트릭트 쇼 청
로드 넘버 8 테크놀로지 포춘 센터 빌딩 에이 8층
톈, 준
중국 100192 베이징 하이 디안 디스트릭트 쇼 청
로드 넘버 8 테크놀로지 포춘 센터 빌딩 에이 8층
(뒷면에 계속)
(74) 대리인
백만기, 양영준, 전경석

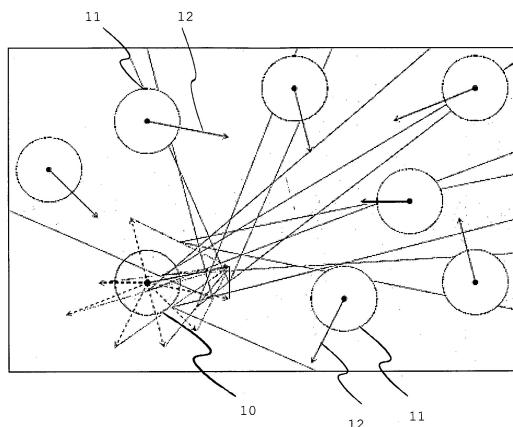
전체 청구항 수 : 총 15 항

(54) 발명의 명칭 **트레이싱 정보에 기초한 충돌 회피 방식을 사용하여 캐릭터들을 애니메이팅하기 위한 방법**

(57) 요 약

그룹 내의 캐릭터에 대해 이동 방향 또는 이동 속도를 결정하기 위한 방법으로서, 상기 캐릭터가 위치한 영역 맵에서의 셀로부터 트레이싱 정보를 판독하는 단계, 충돌 회피가 필요한지를 판단하는 단계 및 만약 충돌 회피의 움직임이 필요하다면 현재의 영역 셀에서 상기 트레이싱 정보를 생성하는 단계를 포함한다.

대 표 도 - 도1



(72) 발명자

카이, 캉잉

중국 100192 베이징 하이 디안 디스트릭트 쉐 칭
로드 넘버 8 테크놀로지 포춘 센터 빌딩 에이 8층

첸, 지보

중국 100192 베이징 하이 디안 디스트릭트 쉐 칭
로드 넘버 8 테크놀로지 포춘 센터 빌딩 에이 8층

특허청구의 범위

청구항 1

그룹 내의 캐릭터에 대해 이동 방향 및 이동 속도 중 적어도 하나를 결정하기 위한 방법으로서,

상기 캐릭터가 위치한 영역 맵(terrain map)에서의 현재의 셀로부터 트레이싱 정보(tracing information)를 판독하는 단계;

충돌 회피가 필요한지를 판단하는 단계;

만약 충돌 회피가 필요하지 않다면, 상기 캐릭터의 종전의 방향 및 종전의 속도(velocity)에 따라 상기 캐릭터를 전진시키는 단계;

만약 충돌 회피가 필요하다면, 매칭되는 방향 및 속도 중 적어도 하나를 탐색하는 단계;

만약 매칭되는 방향 및 속도 또는 이들 중 어느 하나가 발견되면, 충돌시간(time-to-collision)을 계산하고, 상기 계산한 충돌시간을 사전 결정된 범위와 비교하는 단계;

만약 상기 충돌시간이 상기 사전 결정된 범위 내에 있다면, 상기 발견한 매칭되는 방향 및 속도 또는 이들 중 어느 하나에 따라 상기 캐릭터를 전진시키는 단계; 및

만약 상기 매칭되는 방향 및 속도 또는 이들 중 어느 하나가 발견되지 않거나 상기 충돌시간이 상기 사전 결정된 범위 밖에 있다면, 충돌 회피를 위한 움직임(manoeuvre for collision avoiding)을 수행하고, 상기 영역 맵에서의 상기 현재의 셀에서 상기 트레이싱 정보를 갱신하는 단계

를 포함하는 방법.

청구항 2

제1항에 있어서,

상기 영역 맵의 셀은 $d < \Delta t * V^{pref}$ 을 따르는 지름 d를 가지며, Δt 는 시뮬레이션 시간 간격이고 V^{pref} 는 캐릭터들의 임의의 주어진 기본설정 속력(preference speed)인, 방법.

청구항 3

제1항 또는 제2항에 있어서,

복수의 캐릭터를 사용하며, 상기 캐릭터들의 임의의 주어진 기본설정 속력이 모든 캐릭터들에 대해서 동일한, 방법.

청구항 4

제1항 내지 제3항 중 어느 하나의 항에 있어서,

상기 판독된 셀의 트레이싱 정보는 일반적으로 상기 그룹의 행동이 아닌 상기 셀에 들어오는 적어도 다음번 캐릭터의 개별적인 행동에 영향을 주는, 방법.

청구항 5

제1항 내지 제4항 중 어느 하나의 항에 있어서,

상기 충돌 회피가 필요한지를 판단하는 단계는, 캐릭터가 새로운 셀에 들어와서 상기 캐릭터의 진입각(entry angle)을 포함하는 각도의 범위와 적어도 매칭되는 트레이싱 정보를 탐색하는 것을 포함하는, 방법.

청구항 6

제1항 내지 제5항 중 어느 하나의 항에 있어서,

영역 맵에서의 셀의 트레이싱 정보는 상기 셀에 들어온 적어도 하나의 종전 캐릭터에 대한 진입각, 새로운 속도

및 충돌시간을 포함하는, 방법.

청구항 7

제1항 내지 제6항 중 어느 하나의 항에 있어서,

셀의 상기 트레이싱 정보는 적어도 네 개의 데이터 세트를 포함하는데, 각 데이터 세트는 상기 캐릭터의 진입 방향(Ve)의 범위와 관련되며, 각 데이터 세트는 새로운 속도(V_{new}) 및 상기 새로운 속도에 기초한 충돌시간 (time-to-collide: TTC)을 적어도 포함하는, 방법.

청구항 8

제1항 내지 제7항 중 어느 하나의 항에 있어서,

상기 영역 맵에 저장된 상기 정보는, 현재의 상황을 기준으로 적어도 다음번 캐릭터에 대한 속력, 속도 및 방향 중 적어도 하나를 결정하기 위한 지침으로써 사용되는 중간 결과(intermediate result)일 뿐인, 방법.

청구항 9

군중을 시뮬레이션하기 위한 방법으로서,

그룹의 진행 방향을 결정하는 단계;

그룹의 진행 방향이 변경될 때 상기 그룹 내의 각 캐릭터에 대한 속도를 조정하는 단계; 및

상기 그룹에 대한 속도와 각 캐릭터에 대한 새로운 속도를 생성하기 위한 국부적인 조정을 조합하는 단계

를 포함하며, 상기 각 캐릭터에 대한 국부적인 조정은 제1항에 따른 방법을 사용하여 생성되는(generated), 방법.

청구항 10

그룹 내의 캐릭터에 대해 이동 방향 및 이동 속도 중 적어도 하나를 결정하기 위한 장치로서,

영역 맵의 셀들을 저장하기 위한 메모리(51);

상기 캐릭터가 위치한 영역 맵에서의 현재의 셀로부터 트레이싱 정보를 판독하기 위한 제1 데이터 판독 수단 (52);

상기 제1 데이터 판독 수단으로부터의 상기 트레이싱 정보에 기초하여, 충돌 회피가 필요한지 아닌지 여부를 판단하고, 상기 판단한 결과를 통지하기(ca, nca) 위한 제1 충돌 검출기(53);

제1 계산기가 충돌 회피가 필요 없다고 통지하면(nca) 상기 캐릭터의 종전의 방향 및 종전의 속도에 따라 상기 캐릭터를 전진시키기 위한 제1 위치 계산기(54);

상기 제1 계산기가 충돌 회피가 필요하다고 통지하면(ca), 매칭되는 방향(dir) 및 속도(vel) 중 적어도 하나를 탐색하기 위한 제1 프로세서(55);

상기 발견한 매칭되는 방향(dir) 및 속도(vel) 또는 이들 중 어느 하나에 기초하여 충돌시간(TTC)을 계산하기 위한 제2 프로세서(56);

상기 계산한 충돌시간(TTC)을 사전 결정된 범위와 비교하기 위한 비교기(57); 및

상기 발견한 매칭되는 방향(dir) 및 속도(vel) 또는 이들 중 어느 하나에 따라 상기 캐릭터를 전진시키고, 상기 매칭되는 방향 및 속도 또는 이들 중 어느 하나를 발견하지 못했거나 상기 충돌시간이 상기 사전 결정된 범위 밖에 있으면, 충돌 회피 움직임을 수행하고 상기 현재의 영역 셀의 상기 트레이싱 정보를 갱신하기 위한 제2 위치 계산기(58)

를 포함하는 장치.

청구항 11

제10항에 있어서,

영역 맵에서의 셀의 트레이싱 정보는 상기 셀에 들어온 적어도 하나의 종전 캐릭터에 대한 진입각, 새로운 속도 및 충돌시간을 포함하는, 장치.

청구항 12

제10항 또는 제11항에 있어서,

셀의 상기 트레이싱 정보는 적어도 네 개의 데이터 세트를 포함하는데, 각 데이터 세트는 상기 캐릭터의 진입 방향(V_e)의 범위와 관련되며, 각 데이터 세트는 새로운 속도(V_{new}) 및 상기 새로운 속도에 기초한 충돌시간(TT_C)을 적어도 포함하는, 장치.

청구항 13

제10항 내지 제12항 중 어느 하나의 항에 있어서,

상기 제1 충돌 검출기(53)는, 새로운 셀에 들어오는 캐릭터에 대해서 충돌 회피가 필요한지를 판단할 때, 상기 캐릭터의 진입각을 포함하는 각도의 범위와 적어도 매칭되는 트레이싱 정보를 탐색하는, 장치.

청구항 14

제10항 내지 제13항 중 어느 하나의 항에 있어서,

상기 영역 맵의 셀은 $d < \Delta t * V^{pref}$ 을 따르는 지름 d 를 가지며, Δt 는 시뮬레이션 시간 간격이고 V^{pref} 는 캐릭터들의 임의의 주어진 기본설정 속력인, 장치.

청구항 15

군중을 시뮬레이션하기 위한 장치로서,

그룹의 진행 방향을 결정하고;

그룹의 진행 방향이 변경될 때 상기 그룹 내의 각 캐릭터에 대한 속도를 조정하고;

상기 그룹에 대한 속도와 각 캐릭터에 대한 새로운 속도를 생성하기 위한 국부적인 조정을 조합하기 위한 프로세서

를 포함하고, 각 캐릭터에 대한 상기 국부적인 조정은 제10항 내지 제14항 중 어느 하나에 따른 장치를 사용하여 생성되는, 장치.

명세서

기술분야

[0001]

본 발명은 캐릭터들을 애니메이팅하기 위한 방법에 관한 것으로, 상세하게는 군중(crowds)이라고도 부르는 대규모의 캐릭터 그룹을 애니메이팅하기 위한 방법에 관한 것이다.

배경기술

[0002]

군중 시뮬레이션은 게임, 영화, 도시계획 및 긴급 피난 시뮬레이션에서 널리 사용한다. 캐릭터들의 그룹 또는 군중은, 함께 모여있는 사람들/캐릭터들의 집단이며, 같은 목적을 가졌고 유사하게 행동한다. 그룹으로의 행동은 우리의 일상 생활에서 매우 일반적이며 널리 연구되고 있다. 만약 군중 내에서 아주 많은 그룹의 움직임이 있다면, 대체로 각 그룹에 대한 경로를 먼저 계획한 다음, 그룹 내에서 각 캐릭터의 활동을 계획하여 시뮬레이션하는 것으로 알려져 있다. 군중 시뮬레이션에서 가장 중요한 측면들 중 하나는 충돌 회피, 특히 가상의 캐릭터들(아바타들) 간의 충돌 회피이다. 캐릭터들이 자연스럽게 보이도록 하기 위해서, 각 캐릭터는 개별적으로 행동해야만 하고, 특히 임의의 주어진 영역(terrain)에서 자신만의 방식으로 이동해야만 한다. 상기 영역은 영역 맵 또는 로드맵(roadmap)으로 나타난다. 문제는, "포텐셜 필드" 움직임("potential field" manoeuvre: 예컨대 그룹 내의 각 캐릭터가 개별적인 위치 및 환경에 따라 이동함) 때문에, 그룹 내의 각 캐릭터에 대해 개별적인 위치 및 상황에 대한 평가가 이루어져야 하기 때문에, 시뮬레이션의 복잡도가 높다는 것이다. "포텐셜 필드" 움직임에서, 각 캐릭터의 위치 및 환경은 다음 단계를 결정하기 위해서 맨 처음부터(from scratch) 계산된

다.

[0003] 예를 들어 J. van den Berg, M. Lin, D. Manocha가 IEEE 로보틱스 및 자동화에 관한 국제회의(International Conference on Robotics and Automation)에서 발표한 "실시간 멀티 에이전트 내비게이션을 위한 상호속도장애물(Reciprocal velocity obstacles for real-time multi-agent navigation)"에서와 같이, 일반적으로 속도 조정에 관한 복잡한 계산이 수행된다.

[0004] [참고문헌]

[0005] 1. J. van den Berg, M. Lin, D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," IEEE International Conference on Robotics and Automation, 2008.

[0006] 2. F. Tecchia, Y. Chrysanthou, "Real-time rendering of densely populated urban environments," Proceedings of the Eurographics Workshop on Rendering Techniques, 2000.

발명의 내용

[0007] 본 발명에 따르면, 적어도 계산된 새로운 속도에 대한 정보가 영역 맵의 셀들 또는 로드맵에 저장될 것이고, 셀에 들어오는 적어도 다음번 캐릭터에 영향을 줄 것이다. 일반적으로 상기 로드맵이 그룹의 행동이 아닌, 적어도 다음번 캐릭터의 개별적인 행동에 영향을 주는 것이 바람직하다. 그리드 탐색(grid searching)의 알려진 기술에서와는 다르게, 상기 정보는 경로 계획에 대해 사용되는 것과 같은 고정된 영역의 장애물과는 관련이 없다. 대신에, 영역 맵에 저장된 정보는 단지 중간 결과이며, 적어도 다음번 캐릭터에 대한 현재의 상황을 기초로 결정하기 위한 지침으로써 사용된다.

[0008] 하나의 특징으로서, 본 발명은 그리드 셀들의 크기를 어떻게 바꿀 것인지, 이를 충돌 회피를 위해 어떻게 사용할 것인지, 정보의 유효함을 어떻게 판단할 것인지를 다룬다.

[0009] 본 발명의 일실시예에서, 그룹 내의 캐릭터에 대해 이동 방향 및 이동 속도 중 적어도 하나를 결정하기 위한 장치로서, 영역 맵의 셀들을 저장하기 위한 메모리, 상기 캐릭터가 위치한 영역 맵에서의 현재의 셀로부터 트레이싱 정보를 판독하기 위한 제1 레이터 판독 수단, 상기 제1 레이터 판독 수단으로부터의 상기 트레이싱 정보에 기초하여, 충돌 회피가 필요한지 여부를 판단하고, 상기 판단한 결과를 통지하기 위한 제1 충돌 검출기, 제1 계산기가 충돌 회피가 필요 없다고 통지하면 상기 캐릭터의 종전의 방향 및 종전의 속도에 따라 상기 캐릭터를 전진시키기 위한 제1 위치 계산기, 상기 제1 계산기가 충돌 회피가 필요하다고 통지하면, 매칭되는 방향 및 속도 중 적어도 하나를 탐색하기 위한 제1 프로세서, 상기 발견한 매칭되는 방향 및/또는 속도를 기초로 충돌시간(time-to-collision)을 계산하기 위한 제2 프로세서, 상기 계산한 충돌시간을 사전 결정된 범위와 비교하기 위한 비교기 및 상기 발견한 매칭되는 방향 및/또는 속도에 따라 상기 캐릭터를 전진시키고, 상기 매칭되는 방향 및/또는 속도를 발견하지 못했거나 상기 충돌시간이 상기 사전 결정된 범위 밖에 있다면, 충돌 회피 움직임을 수행하고 상기 현재의 영역 셀의 상기 트레이싱 정보를 갱신하기 위한 제2 위치 계산기를 포함한다.

[0010] 예를 들어, 임의의 주어진 영역은 정해진 크기의 셀들로 분할된다. 상기 크기는 다양한 파라미터, 예컨대 통상적인 속도에 의해 결정된다(이하 참조). 영역 셀에 진입하는 캐릭터에 대해서, (예를 들어, 북쪽(N), 동쪽(E), 남쪽(S), 서쪽(W) 중 하나 또는 북쪽(N), 북동쪽(NE), 동쪽(E), 남동쪽(SE) 중 하나 등과 같은) 진입 방향이 결정되고, 같은 방향으로 들어갔던 종전 캐릭터들에 대해서 방향 및/또는 속도 판단이 어떠했었는지 결정된다. 이후 상기 판단으로 이어진 임의의 전제조건이 여전히 타당한지를 확인할 수 있다.

[0011] F. Tecchia, Y. Chrysanthou가 렌더링 기술에 대한 유로그래픽 워크숍의 논문집(Proceedings of the Eurographics Workshop on Rendering Techniques)에서 2000년에 발표한 "인구 밀도가 높은 도시 환경에서의 실시간 렌더링(Real-time rendering of densely populated urban environments)"에서 설명한 것처럼, 상호속도장애물(Reciprocal Velocity Obstacle: RVO)은 충돌 회피를 다루는 새로운 기술이다. 두 명의 캐릭터에 대한 시나리오의 경우에서, 충돌을 예방할 수 있도록 캐릭터들이 선택해서는 안 되는 속도들을 나타내는 속도 평면상에 각 캐릭터에 대한 RVO (범위 또는) 영역이 존재하게 된다. 만약 캐릭터가 RVO 영역 밖의 임의의 속도를 선택한다면, 다른 캐릭터와의 충돌을 회피하는 것이 보장된다. RVO 기술은 각 캐릭터가 충돌을 회피하기 위해 같은 방안을 채택한다고 또한 가정한다. 다시 말해, 캐릭터들은 책임을 분담하며 임의의 충돌을 회피하기 위해서 그에 따라 조치를 취한다.

도면의 간단한 설명

[0012]

본 발명의 대표적인 실시예들이 다음의 첨부 도면을 참조하여 설명된다.

도 1은 상호속도장애물에 관한 복수의 캐릭터의 시나리오를 도시하는 도면.

도 2는 시뮬레이션 방법의 단계들을 도시하는 도면.

도 3은 영역 그리드의 일부분을 도시하는 도면.

도 4는 그리드 부분 및 트레이싱 정보를 도시하는 도면.

도 5는 그룹 내의 임의의 캐릭터에 대한 이동 속도 및 이동 방향 중 적어도 하나를 결정하기 위한 장치의 블록도.

발명을 실시하기 위한 구체적인 내용

[0013]

도 1은 RVO에 관한 복수의 캐릭터의 시나리오를 도시한다. 도 1에서, 한 캐릭터(10)는 모든 이웃한 캐릭터의 RVO들의 조합 밖에 놓인 속도를 선택해야만 한다. 이웃한 캐릭터들(11)은 각각 이동 방향 및 속력(12)을 가진다. 그러나 매우 혼잡한 상황에서는, 이러한 속도가 존재하지 않을 수도 있다. 따라서, 그럼에도 불구하고 시뮬레이션을 진행하기 위해서는 캐릭터에 대한 속도를 선택해야만 한다. 그러므로 최소한의 "페널티(penalty)"를 갖는 속도를 선택한다. 캐릭터 i 에 대한 일반적인 페널티 함수는 다음과 같다.

수학식 1

$$\text{penalty}_i(\mathbf{v}'_i) = w_i \frac{1}{tc_i(\mathbf{v}'_i)} + \|\mathbf{v}_i^{\text{pref}} - \mathbf{v}'_i\|$$

[0014]

수학식 1에서, \mathbf{v}'_i 은 새로운 속도이고; $\mathbf{v}_i^{\text{pref}}$ 는 선호되는 속도(preference velocity; 이하 "기본설정 속도"라 함)이며; $tc_i(\mathbf{v}'_i)$ 는 충돌까지의 예상 시간이고; w_i 는 충돌시간과 속도 전환 간의 균형을 맞추기 위한 가중치이다. $\|\dots\|$ 은 거리를 표시한다.

[0016]

이러한 기술은 실제로 좋은 성능을 나타내지만, 계산의 복잡도가 너무 높아서 많은 수의 캐릭터를 갖는 군중을 시뮬레이션하는 것은 불가능하다. 매시간 단계, 각 캐릭터에 대해서, 시뮬레이터는 수백 개의 가능 속도(admissible velocities)를 샘플링할 필요가 있고, 최소한의 페널티를 갖는 속도를 찾아내기 위해서 각 속도에 대한 페널티를 계산한다 (이후, 이 프로세스를 "RVO 속도 선택"이라 한다). 그러므로 "속도 선택"의 복잡도를 어떻게 줄이느냐는 것이 대규모 군중 시뮬레이션에서, 특히 실시간 애플리케이션용으로 매우 중요하다.

[0017]

다음은 새로운 속도 선택 방법을 제안한다. "RVO 속도 선택"과 비교해서, 하기의 방법은 군중 시뮬레이션의 총 복잡도를 상당히 줄인다.

[0018]

제안된 방법은 대상 캐릭터에 대한 새로운 속도를 선택하기 위해 종전 캐릭터들의 트레이싱 정보를 이용함으로써 군중 시뮬레이션의 복잡도를 줄이게 된다. 먼저, 캐릭터들이 걷는 (예를 들어, 2D 평면과 같은) 평면을 정규 그리드(regular grid)로 이산화한다. 상기 그리드의 해상도(resolution)는 기본설정 속도 및 시뮬레이션의 시간 단계에 의해서 결정된다. 트레이싱 정보는 진입각(entry angle), 새로운 속도 및 충돌시간을 포함하며, 상기 그리드의 각 셀에 저장된다. 이후의 캐릭터가 새로운 셀에 들어가게 되면, 캐릭터의 진입각과 매칭되는 (즉, 실질적으로 같은 진입각을 갖는) 트레이싱 정보를 탐색하고, 새로운 속도를 기초로 충돌시간을 계산한다. 만약 현 캐릭터에 대한 충돌시간이 상기 셀에 저장된 것과 필적한다면 (적어도 만약 이 둘이 같다면), 새로운 차례(round)의 RVO-스타일의 충돌 회피 계산을 하지 않고 캐릭터는 새로운 속도를 선택한다; 그렇지 않으면, 기존의 RVO 충돌 회피를 수행하며, 상기 셀의 트레이싱 정보는 그에 따라 갱신된다.

[0019]

도 2는 제안된 시뮬레이션 체계의 단계들을 도시하고 있다. 이하 각 단계에 대해 자세히 설명한다.

[0020]

준비 단계(20)에서, 이산화 및 초기화를 수행한다(P0). 시뮬레이션 이전에, 영역은 셀들의 정규 그리드로 이산화된다 (이를 3D 영역의 표면에 쉽게 적용할 수 있었으나, 본 명세서에서는 단순화를 위해 2D 평면을 사용한다). 도 3은 해상도라 부르는 지름 d 를 갖는 영역 그리드의 부분을 도시한다. 그리드의 해상도는, 각 시간 단계 동안 캐릭터가 기본설정 속력(preference speed)으로 새로운 셀로 걸어가야만 한다는 원칙에 의해서 결정된

다. 그러므로

수학식 2

$$d < \Delta t * v^{\text{pref}}$$

[0022] 가 되며, 여기서 d 는 (도 3을 참조하면) 그리드 셀의 대각선 길이이고, Δt 는 시간 단계의 간격이며, v^{pref} 는 기본설정 속력이다. 일실시예에서, 단순화를 위해 모든 캐릭터에 대해 이는 동일하다고 가정한다.

[0023] 도 4는 그리드 부분 및 트레이싱 정보를 도시한다. 그리드 셀에 저장된 트레이싱 정보는 적어도 네 개의 데이터 세트를 포함하며, 각각은 캐릭터의 진입 방향(V_e)의 범위와 관련이 있다. 각 데이터 세트에는, 새로운 속도(V_{new}) 및 이 새로운 속도에 기초한 충돌시간(time-to-collide: TTC)이 포함되어 나타나 있다.

[0024] 도 2에 도시된 제2 단계에서, 시뮬레이션 프로세스를 수행한다. 만약 충돌 회피가 필요하다면, 제1 모듈(P1)은 탐색 단계(22)에서 매칭되는 대상에 대한 탐색을 수행한다. 즉, 충돌 회피 움직임이 필요하다고 판단할 때마다 (21), 이 캐릭터의 진입 방향과 매칭되는 트레이싱 정보가 존재하는지 판단하기 위해서(23) 상기 캐릭터에 의해 점유되고 있는 셀을 탐색한다(22). 만약 상기 트레이싱 정보가 존재하지 않는다면, 보통의 충돌 회피 움직임을 수행하며(27), 이는 공지의 것으로서 대개 매우 복잡하다.

[0025] 다음 단계(24)에서, 제2 모듈(P2)은 충돌시간(time-to-collision: TTC)을 계산한다. 캐릭터가 트레이싱 정보로부터 새로운 속도를 얻었다고 가정해도, 여전히 상기 캐릭터는 다른 캐릭터들과 충돌할 가능성이 있다. TTC는 첫 번째 충돌이 발생하는데 걸리는 시간이다. 만약 충돌이 발생하지 않을 것이라고 판단되면(25), TTC는 충분히 커지게 된다. 이러한 계산된 TTC는 다음과 같은 상태에 있게 되는데,

$$TTC^{\text{computed}} > \max(TTC^{\text{tracing}} - k * \Delta t, \Delta t)$$

[0027] 여기서 TTC^{tracing} 은 셀에 저장된 트레이싱 정보이다. 이와 달리 매칭되지 않으면, 보통의 충돌 회피 움직임이 수행될 것이다(27). 상수 k 는 시뮬레이션의 시간 단계들의 수를 나타내며, 불일치의 허용 오차(tolerance of mismatch)를 나타내기 위해서 적절하게 선택되어야만 한다. 그 수가 작을수록, 매칭이 더욱 정밀해진다. 매칭의 정밀성과 매칭이 성공할 가능성 간에는 트레이드 오프(trade-off)가 존재한다. 또한, 만약 k 가 지나치게 크면, 매칭되는 트레이싱 정보가 현 캐릭터에 지침이 되기 어려우며, 따라서 부적절한 새로운 속도로 이어져 충돌 회피가 더욱 비효율적으로 될 수 있다. k 의 실제 값은 임의의 주어진 간격의 시뮬레이션 시간 단계(Δt), 즉 시뮬레이션의 시간 해상도(temporal resolution)에 크게 의존한다.

[0028] 제3 모듈(P3)은 보통의 충돌 회피 움직임을 수행한다(27). 만약 셀에 트레이싱 정보가 존재하지 않거나 사용 가능한 트레이싱 정보가 잘 매칭되지 않으면, 현 캐릭터에 대한 새로운 속도를 선택하기 위해서 보통의 트레이싱 회피 움직임이 수행될 것이다. 한 가지 제안으로, 상기 언급한 "실시간 멀티 에이전트 내비게이션을 위한 상호속도장애물"이란 참고문헌에서처럼, 속도의 선택 문제를 다루기 위해서 공지의 RVO 충돌 회피 기술을 사용하는 것이다.

[0029] 제4 모듈(P4)은 영역 셀에서 트레이싱 정보를 생성한다(28). 보통의 충돌 회피 움직임이 수행될 때마다, 셀에 저장된 해당 트레이싱 정보는 생성되어야만 한다.

[0030] 제5 모듈(P5)은 캐릭터를 전진시킨다(26). 새로운 속도에 기초하여, 예를 들어 뉴턴 역학을 사용함으로써 새로운 위치가 캐릭터에 대해 계산된다 (즉, 거리 s , 속도 v 및 시간 t 를 사용하여 $s=v*t$ 에 의해서 새로운 위치가 계산된다). 새로운 위치는 영역 맵 또는 별도의 메모리에 저장된다.

[0031] 일실시예에서, 본 발명은 그룹 내의 임의의 캐릭터에 대해 이동 방향 및 이동 속도 중 적어도 하나를 결정하기 위한 장치에 관한 것이다. 도 5는 이와 같은 장치의 블록도를 도시한다. 그룹 내의 임의의 캐릭터에 대해 이동 방향 및 이동 속도 중 적어도 하나를 결정하기 위한 장치는, 영역 맵의 셀들을 저장하기 위한 메모리(51), 상기 캐릭터가 위치한 영역 맵에서의 현재의 셀로부터 트레이싱 정보를 판독하기 위한 (예를 들어 메모리 접근 장치와 같은) 제1 데이터 판독 수단(52), 상기 제1 데이터 판독 수단으로부터의 상기 트레이싱 정보에 기초하여, 트레이싱 회피가 필요한지 여부를 판단하고, 판단한 결과를 통지하기(ca, nca) 위한 제1 충돌 검출기 (53), 제1 계산기가 충돌 회피가 필요 없다고 통지하면(nca), 상기 캐릭터의 종전의 방향 및 종전의 속도에 따

라 상기 캐릭터를 전진시키기 위한 제1 위치 계산기(54), 상기 제1 계산기가 충돌 회피가 필요하다고 통지하면, 매칭되는 방향(dir) 및 속도(vel) 중 적어도 하나를 탐색하기 위한 제1 프로세서(55), 상기 발견한 매칭되는 방향 및/또는 속도에 기초하여, 충돌시간(TTC)을 계산하기 위한 제2 프로세서(56), 상기 계산한 충돌시간(TTC)을 사전 결정된 범위와 비교하기 위한 비교기(57) 및 상기 발견한 매칭되는 방향 및/또는 속도에 따라 상기 캐릭터를 전진시키고, 만약 매칭되는 방향 및/또는 속도가 발견되지 않거나 충돌시간이 사전 결정된 범위 밖에 존재한다면, 충돌 회피 움직임을 수행하고 현재의 영역 셀에서 상기 트레이싱 정보를 갱신하기 위한 제2 위치 계산기(58)를 포함한다.

[0032] 본 발명은 다른 애플리케이션 시나리오에서 사용할 수 있다. 일반적으로, 본 방법은 보통의 충돌 회피 움직임의 시간을 줄여서, 군중 시뮬레이션의 복잡도를 줄인다. 영역 셀 상에 종전에 존재했던 다른 캐릭터와 유사한 진입 방향을 갖는 현 캐릭터에 대해서, 상기 영역 셀에 저장된 트레이싱 정보를 지침으로 사용함으로써, 복잡한 계산을 하지 않고 바로 저장된 속도를 사용할 수 있었다.

[0033] 이러한 방법은, 다른 캐릭터들을 뒤따르는 그룹 내의 캐릭터들이 종전의 캐릭터들과 유사한 상황에 직면할 것이라는 가정에 기초한다. 이는, 두 그룹의 캐릭터들이 서로 교차할 때 특히 들어맞는다. 따라서, 본 발명의 방법은 다른 그룹의 길을 횡단하는 그룹 또는 다른 그룹과 마주치는 그룹과 같이, 서로 간섭을 일으키는 그룹들 내 각 캐릭터의 운동 방향 및 속도를 결정하기 위해 사용할 수 있다. 상기 언급한 가정이 정당하기만 하면, 예컨대 영화에서의 격투 장면, 교통 시뮬레이션, 대피 시뮬레이션 등과 같은 임의의 군중 시뮬레이션에서 본 방법을 사용할 수 있다.

[0034] 제안된 방법은 (수천 또는 그 이상의 캐릭터처럼) 대규모 군중을 시뮬레이션하는 효과적인 방법이다. 시뮬레이션되는 군중의 행동을 제어하는 몇몇 입력 파라미터가 존재한다. 한 가지는 d (즉 그리드의 해상도의 지표)로서, 이는 움직임의 부드러움을 결정한다; 다른 것으로는 TTC 공식에서의 k 로, 이는 불일치의 허용 오차를 결정한다. 예를 들어, 본 발명을 사용하는 시뮬레이션 소프트웨어에서, 이러한 파라미터들은 프로그램될 수 있다. 본 발명은 영역 그리드 셀들 내에서의 뚜렷한 "추종(following)" 패턴들을 이끄는데, 이는 뒤따르는 캐릭터들이 종전 캐릭터들로부터의 트레이싱 정보를 사용하기 때문이다. 즉, 서로 뒤따르는 캐릭터들은 일반적으로 같은 길로 갈 것이지만, (예컨대, 이동하는 방해물과 같은) 현 상황을 고려하면 실제 인간 군중의 행동과 유사하게 된다.

[0035] 본 발명은, 예를 들어 먼저 각 그룹에 대한 경로 및 속도를 계획하고 이후 그룹 내의 각 캐릭터에 대한 속도를 조정하는 것에 의해서, 군중을 시뮬레이션하는데 사용할 수 있다. 단계들은 다음과 같다.

[0036] 1. 그룹들에 대한 전역 계획(global planning) 및 국부적인 충돌 회피 단계; 각 그룹은 단일의 개체로 취급된다 (즉, 그룹의 진행 방향을 각 그룹에 대해서 결정한다).

[0037] 2. 상기 그룹의 진행 방향이 바뀔 때마다 상기 그룹 내의 각 캐릭터의 속도를 조정하는 단계.

[0038] 3. 상기 그룹에 대한 속도와 각 캐릭터에 대한 새로운 속도를 생성하기 위한 국부적인 조정을 조합하는 단계.

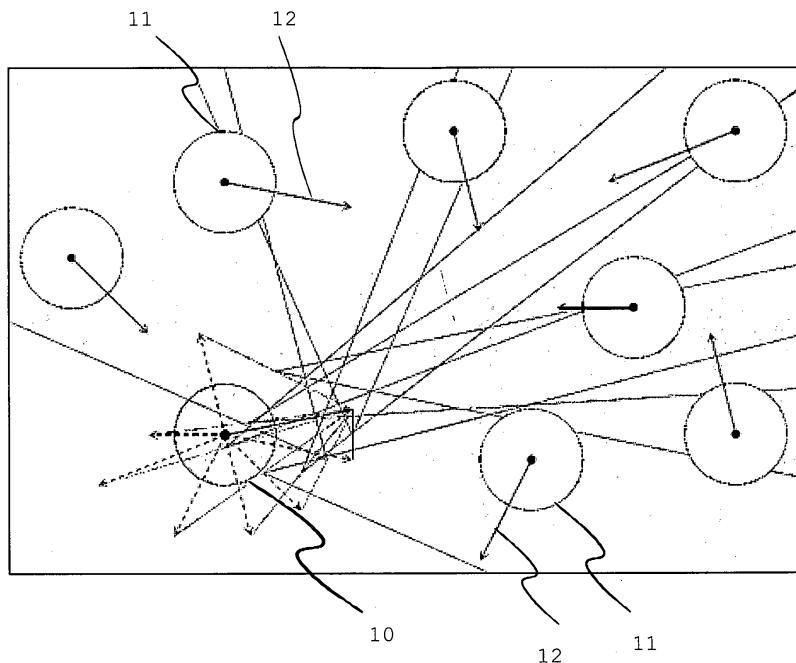
[0039] 다른 캐릭터들을 뒤따르는 캐릭터들에 대해서, 반드시 완전한 계산을 다시 수행할 필요는 없다. 대신에 계산한 새로운 위치에 대한 정보는 영역 맵의 셀들에 저장될 것이다. 본 방법은 군중 시뮬레이션의 복잡도를 줄이며, 영화 또는 게임 제작, 특히 군대와 같은 일관성 있는 캐릭터들의 그룹의 행동을 시뮬레이션할 때 사용할 수 있다.

[0040] 본 발명의 바람직한 실시예들에 적용된 것처럼 본 발명의 근본적인 새로운 특징들을 도시하고, 설명하며, 지적하였으나, 본 발명의 정신으로부터 벗어나지 않고도 본 발명이 속하는 기술분야에서 통상의 지식을 가진 사람들에 의해서, 설명한 장치 및 방법에서의 다양한 생략, 치환 및 변경이 개시된 장치의 형태 및 세부사항에서, 장치의 동작에서 이루어질 수 있음을 이해해야 할 것이다. 동일한 결과를 달성하기 위해서 실질적으로 동일한 방법으로 실질적으로 동일한 기능을 수행하는 요소들의 모든 조합이 본 발명의 범위 내에 존재하도록 명백히 의도하였다. 한 가지 설명한 실시예로부터 다른 것으로의 요소의 치환을 또한 충분히 의도하고 고려할 수 있다.

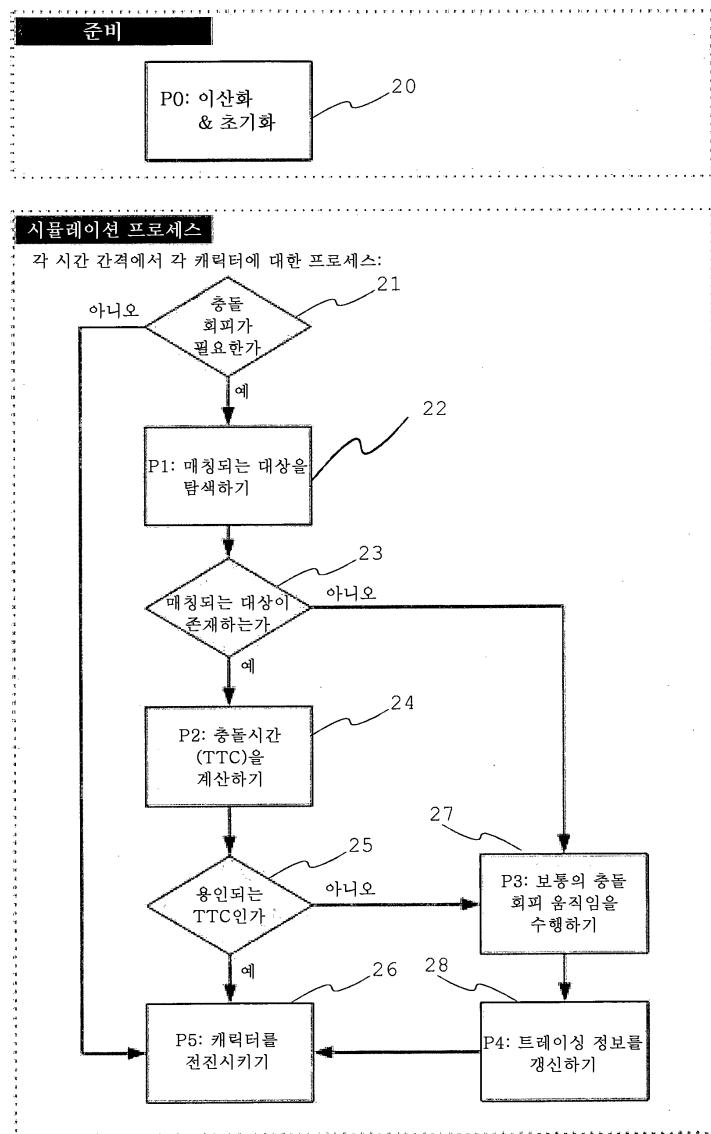
[0041] 본 발명은 오직 예시로써 설명되었으며, 본 발명의 범위로부터 벗어나지 않고도 세부사항의 변경을 할 수 있음을 이해할 것이다. 상기 설명 및 (적절한 위치의) 청구항 및 도면에서 개시된 각 특징은 독립적으로 또는 임의의 적절한 조합으로 제공될 수 있다. 청구항에 나타난 참조 번호들은 단지 예로써 존재하며, 청구항의 범위에 아무런 제한 효과를 미치지 않는다.

도면

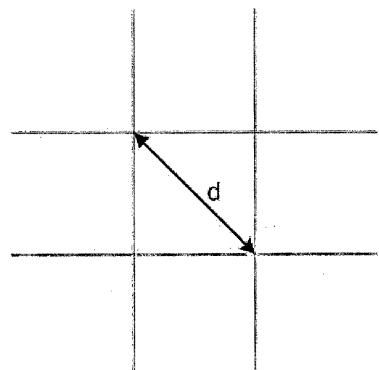
도면1



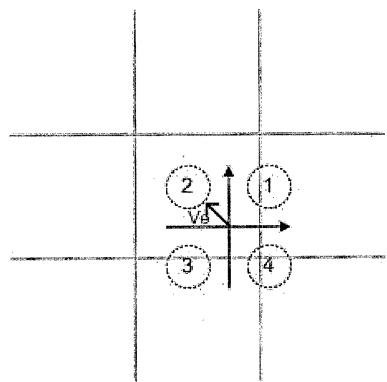
도면2



도면3



도면4



도면5

