

실시간 스프레이 페인팅 시뮬레이션에 관한 연구

김대석[○] 박진아

컴퓨터 그래픽스 및 가시화 연구실, 한국정보통신대학교

greatstone@icu.ac.kr, jinah@icu.ac.kr

Real-time Spray Painting Simulations

Daeseok Kim[○] Jinah Park

Computer Graphics and Visualization Lab, Information and Communications University

본 논문에서는 가상환경에서 페인트를 분사하여 실시간으로 물체를 도색하는 시뮬레이션을 위한 충돌 처리 및 시각화 알고리즘을 제시한다. 이를 통하여 물체에 페인트가 뿌려지면서 도색 되는 모습을 사실적으로 표현해 줄 뿐만 아니라, 페인트 누적 모델을 이용하여 물체에 누적된 페인트의 두께 정보까지 시뮬레이션 하여 시각화함으로써 가상현실 시스템에 적용할 수 있도록 한다. 분사되는 유체 시뮬레이션을 위해서 기존에는 파티클 시스템이 이용되고 있으나 실시간으로 도색이 되는 과정을 시각화하기 위해서는 수백만 개의 파티클에 대하여 충돌 검사를 수행해야 하기 때문에 적절하지 않다. 또한, 파티클 시스템을 이용해 페인트 분사 가시화를 하게 될 경우, 파티클의 입자의 크기 및 운동 모델, 점성 등 고려해야 할 요소가 상당히 많고 사실적인 묘사를 위해 이들의 값을 결정하는데 많은 어려움이 따른다. 따라서 본 논문에서는 스프레이 분사를 통한 페인트 도색을 좀 더 간단한 방법으로 실시간으로 시뮬레이션 하기 위한 새로운 알고리즘을 제안한다.

페인트 시뮬레이션을 위해 본 연구에서는 실제 스프레이 페인팅 실험 결과 데이터에 근거한 가우시안 페인트 누적 모델을 정의한다. 페인트 누적 모델은 페인트가 분사 될 때 분사된 지점으로부터 어느 정도 떨어진 한 지점에 페인트 입자들이 도달할 확률이나 이곳에 물체가 존재하여 표면에 달라붙었을 경우의 페인트의 두께를 계산하기 위한 모델로 주어진 페인트 종류와 스프레이 기구의 종류에 따라 모델링한다. 따라서 정의한 모델을 이용하여 페인트가 분사된 영역에 도색되는 페인트의 두께를 계산할 수 있을 뿐만 아니라, 페인트가 도색되는 영역인지를 판별하기 위한 유효성 검사도 수행할 수 있다.

도색 영역을 찾는 방법으로 본 논문에서는 페인트 누적 모델을 이용한 휴리스틱 알고리즘을 제안한다. 알고리즘의 흐름은 다음과 같다. 첫째, 페인트가 분사되는 영역을 파티클 대신 몇 개의 직선으로 대체하여 이를 이용해 물체들과 충돌 검사를 수행한다. 이 때 페인트가 점에서 일정 각도의 범위로 분사된다고 할 때 이 범위 안에서 임의로 직선을 선택하였다. 둘째, 충돌 검사를 통해 찾은 점들에서 시작하여 주변 점들을 확장 탐색하여(flood-fill algorithm) 한 면에서의 도색 영역을 픽셀 단위로 모두 찾는다. 이 때 도색 영역인지 아닌지를 판단하기 위해 페인트 누적 모델을 이용하게 된다. 셋째, 도색 영역이 한 면을 넘어가게 되는 경우에는 새로운 직선을 생성하여 이웃한 면을 찾는다. 한 면의 가장자리까지 탐색을 하게 되었다면 페인트가 그 위치까지 칠해지거나 그 뒤쪽의 면에 더 칠해지거나 두 가지 경우가 있을 수 있다. 따라서 분사 지점에서 그 가장자리 바깥쪽으로 향하는 새로운 직선을 생성해 충돌검사를 통해 다른 면에도 칠해지는 지를 점검한다. 넷째, 탐색된 도색 영역 중 다른 면에 가려져 도색이 되지 않아야 하는 부분을 제거한다. 확장을 통한 탐색의 문제점은 충돌검사를 끊임없이 수행하지 않고 페인트 누적 모델을 이용해 유효성을 판단하기 때문에 앞에 다른 면이 가려졌을 경우에도 유효하다고 판단하게 된다는 것이다. 따라서 이 부분을 제거하기 위한 방법으로 충돌검사를 위해 만들었던 직선을 이용하여 범위를 재설정한다. 도색 영역을 모두 찾은 이후에 페인트 효과를 적용한다. 이 때 분사된 페인트는 중력이나 바람의 영향을 받지 않고 직선 운동을 한다고 가정하였다.

위에서 제안한 알고리즘을 펜티엄3-3.4 GHz, 3.12GB RAM, NVIDIA GeForce6600, Windows XP의 Visual Studio .NET 환경에서 OpenSG 라이브러리를 사용하여 구현하였다. 실제 선박의 모델과 분사 도구 모델을 위해 선박을 제조할 때 기본 단위로 만들어지는 모듈을 실제 모습, 실제 크기와 동일하게 모델링 하였다. 페인팅 효과를 위해서 텍스처 맵의 정보를 수정하였다. 또한, 페인트가 누적되는 효과를 주기 위해 텍스처 맵과 같은 크기의 페인트 두께 맵을 생성해두고, 페인트 누적 모델을 통해 계산된 두께 값을 두께 맵에 가산하였다. 페인트 효과만을 시각화 했을 때는 일정 이상 페인트를 계속 뿌려도 두께가 증가하는 것을 알 수가 없기 때문에 텍스트나 그래프 그리고 페인트가 칠해진 면을 두께에 따라 다르게 색을 지정하는 방법으로 페인트의 두께를 시각화 하였다.

페인트 시뮬레이션을 위한 알고리즘의 시간 복잡도를 분석해 보면, 초기 정의된 직선의 수를 R, 탐색 과정 중 생성되는 직선의 수를 P, 충돌 검사를 하는 시간을 C, 전체 도색이 되는 텍셀의 수를 T라고 했을 때, 한 프레임을 계산할 때 걸리는 수행 시간은 $(R+P)C + 2T$ 가 된다. 여기서 R+P는 페인트 되는 폴리곤의 수와 비례하고 C는 전체 환경의 폴리곤 수와 비례한다고 할 수 있다. 즉, 페인트 되는 폴리곤의 수와 전체 환경의 폴리곤의 수, 그리고 도색 되는 텍셀의 수에 수행 시간이 영향을 받는다는 것을 알 수 있다. 수행 속도 비교를 위해 폴리곤이 2272개인 환경에서 파티클을 이용한 페인트 시뮬레이션과 본 연구에서 구현한 시뮬레이션의 수행 속도를 측정하였다. 파티클을 이용한 페인트 시뮬레이션에서는 파티클의 수가 500개 일 때 초당 19프레임(fps), 1000개 일 때 10프레임, 1500개 일 때 7프레임의 수행속도를 보였다. 본 연구의 시뮬레이션은 생성된 직선의 수가 100개 일 경우에 대하여 500개의 텍셀에 페인트 되었을 때는 초당 52프레임, 1000개의 텍셀에 페인트 되었을 때는 45프레임, 1500개의 텍셀에 페인트 되었을 때는 40프레임의 수행 속도를 보였다. 여기서 페인트 되는 텍셀의 수는 텍스처의 해상도나 페인트 분사 도구와 면과의 거리에 영향을 받는다. 같은 영역에 페인트를 뿌려도 텍스처의 해상도가 높으면 많은 수의 텍셀에 페인트가 칠해진다. 또한, 페인트 분사 도구가 벽면과 가까운 곳에서 분사를 하면 페인트가 좁은 영역에 칠해지고, 먼 곳에서 분사를 하면 페인트가 넓은 영역에 칠해지게 된다. 직선의 수는 페인트 되는 폴리곤의 복잡도에 따라 달라진다. 페인트 되는 영역이 많은 수의 폴리곤으로 구성되어 있다면 그 폴리곤들을 찾기 위해 많은 직선을 새로 생성하게 된다. 그러나 페인트 되는 영역의 폴리곤 수가 100개가 넘는 복잡한 부분이 아니라면 위의 결과처럼 텍셀이 1500개 이상 페인트 되어도 실시간으로 수행된다는 것을 알 수 있다.

광범위한 영역으로 분사하는 페인트의 실시간 시뮬레이션은 아직 많은 연구가 이루어지지 않았다. 본 논문에서는 광범위로 분사되는 페인트가 표면에 도색 되는 시뮬레이션을 기존 파티클 시스템에 비해 빠른 속도로 수행할 수 있고, 사실적으로 시각화 할 수 있는 새로운 방법을 제시하였다. 또한 수행 속도를 측정하여 실제로 실시간 시뮬레이션이 가능하다는 것을 확인 하였다. 추후 연구 방향으로 본 논문에서는 페인트가 단순히 직선으로 날아가는 상황을 고려하였지만, 바람이 불어 페인트가 날리는 경우나 날아갈 때 중력의 영향을 받는 경우에 대해 생각해 볼 수 있다. 그리고 페인트는 유체이기 때문에 벽에 너무 많은 양이 묻게 되면 흘러내리게 되는데 이를 표현할 수 있는 방법을 고민함으로써 좀 더 사실적인 페인트 시뮬레이션을 만들어 낼 수 있을 것이다.

