본 논문에서 소개하는 파티클 충돌처리 시스템 알고리즘은 파티클의 충돌과 업데이트를 담당하는 단계와 실질적으로 렌더링을 하는 단계 2개의 그래픽스 파이프라인의 구조를 가지고 있다.

첫 번째 단계에선 파티클의 움직임을 업데이트하고 3차원 월드 공간 위치를 2차원 이미지에 이미지에 투영하고 충돌을 처리하는 단계이다.

이 단계의 첫 프레임에는 CPU를 통해 전달되는 정점 데이터를 사용하여 진행되는데 렌더링 결과를 저장한 충돌 탐지 텍스처가 만들어지기 전이기 때문에 빈 텍스처와의 처리가 진행되어 충돌이 일어나지 않는다. 기하 셰이더에서 Transform Feedback 기능을 통해 3차원 위치가 업데이트된 파티클의 데이터가 Transform Feedback Buffer Object에 작성된다. 프래그먼트 셰이더를 통해 3차원 파티클의 위치는 컬러 코드로 변환되어 충돌 감지 텍스처에 업데이트된다.

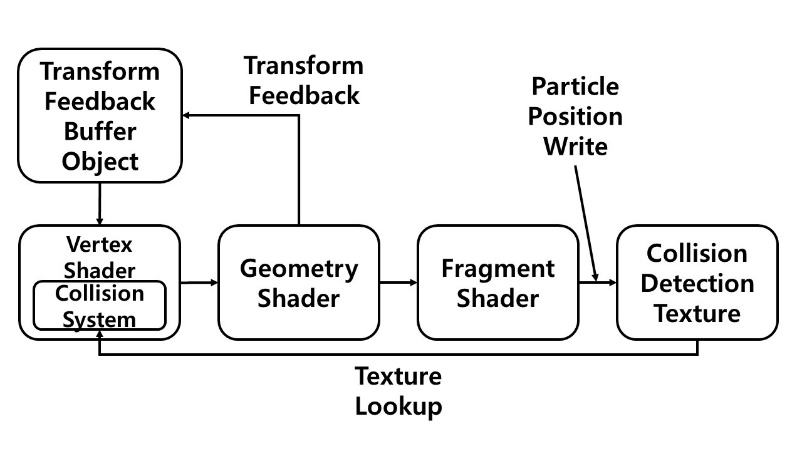


Figure 3 Collision Pass

Figure 3는 충돌 감지 텍스처가 생성된 이후의 충돌처리 단계를 나타낸다.

파티클의 충돌 처리를 위해서 현재 그래픽스 파이프라인의 입력된 정점의 텍셀 값을 계산하여 사용한다. Transform Feedback Buffer Object로 입력된 정점의 3차원 위치 값에 뷰 프로젝션 행렬을 곱하여 -1부터 1사이의 값을 가지는 화면 공간 좌표로 변경한다. 변경된 값의 x, y좌표에 1을 더하고 0.5로 나누어 0부터 1사이의 정규화된 값을 가지는 텍스처 좌표를 얻는다.

얻어진 텍스처 좌표의 주변 픽셀을 순환하여 주변 파티클의 3차원 정보 값을 얻어 충돌을 처리하기 위해 정규화된 값을 다시 역정규화 시켜 반복문을 통한 순환이 용이하도록 변경한다. 역정규화를 위해 텍스처 좌표 값에 충돌 탐지 텍스처의 가로, 세로 해상도 값을 곱해 현재 파티클의 텍셀 좌표를 구한다. 얻어진 텍셀 값을 Base Texel이라 칭하고 Base Texel 값을 이용해 texelFetch 함수를 사용하여 충돌 감지 텍스처에 저장된 현재 파티클의 3차원 위치 정보를 읽어온다.

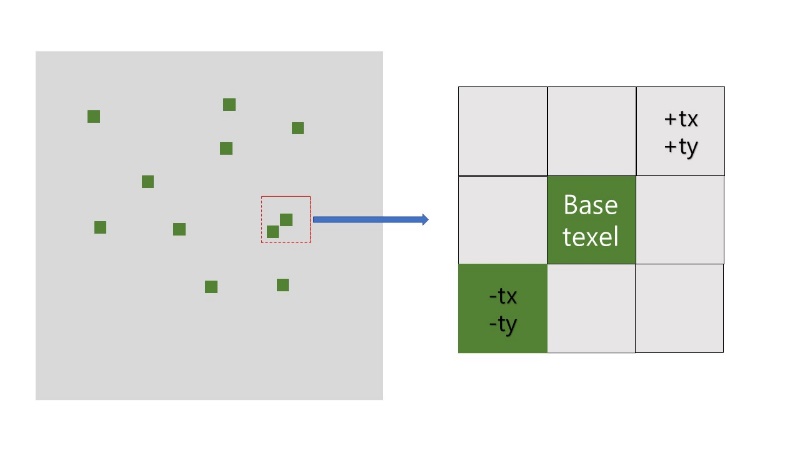


Figure 4 Collision Range

사전에 설정해둔 충돌 탐색 범위 값을 texel Collide Range로 칭하고 Base Texel을 기준으로 tx, ty의 두 값을 -texel Collide Range 부터 texel Collide Range 까지의 범위로 설정하여 값을 1씩 증가시키며 반복문을 통해 Base texel 좌표에 tx, ty 값을 더하여 주변 텍셀에 저장 되어있는 파티클의 3차원 위치 정보를 계산한다. 이때 tx, ty의 두 값이 0을 가지는 경우 충돌 검사를 진행하지 않는다. Figure 4는 충돌 탐색 거리를 1로 설정했을 때의 충돌 탐색 범위를 보여주는 그림이다.

주변 텍셀에서 얻어진 파티클의 3차원 위치 정보와 현재 그래픽스 파이프라인의 입력된 파티클의 3차원 위치 정보 두 값의 직선거리를 계산하여 사전에 설정해둔 충돌 거리 값 보다 작고 충돌 거리의 최소값을 저장하기 위한 particle min length 값 보다 작은 경우를 충돌로 판정한다. 충돌로 판정된 경우 particle min length의 값을 현재 계산된 length의 값으로 갱신하고 충돌된 파티클의 위치를 저장한다.

충돌 탐색 거리를 모두 순회하여 얻어진 length 값과 파티클의 3차원 위치를 사용하여 업데이트를 진행한다. 충돌된 파티클의 속도를 20% 감소시키고, 방향은 두 파티클의 위치의 뺄셈을 통해 얻어지는 벡터를 정규화한 값을 사용한다. 변경된 속도와 방향을 적용하여 3차원 위치 정보를 변경한다. 또한 충돌된 파티클의 시각적 표시 및 지속적인 충돌 방지를 위해 충돌시간을 업데이트한다. 충돌하지 않은 파티클들은 중력의 영향을 받아 업데이트 되어 위치 정보가 변경된다.

파티클의 업데이트된 위치, 방향, 속도, 충돌시간 값을 정점 셰이더의 output으로 기하 셰이더에 전달하여 Transform Feedback Buffer Object에 기록한다. 기록된 데이터 버퍼를 다음 프레임에 입력데이터로 사용한다. 프래그먼트 셰이더에서는 갱신된 3차원 정보 값을 Color 값으로 x,y,z를 작성하여 충돌 감지 텍스처에 갱신한다. 이 과정을 의사코드로 표현하면 다음과 같다.