# AR/VR/XR assignment visualize 2

M2023067 김의찬

본 보고서는 완성본이 아닌 시간상 단순 결과를 도출 후 간단히 visualize image과 간단한 구현 내용을 설명함. 추후 보고서 영어 번역 및 정리를 진행 할 예정

1. Z축 소실점 찾기.  
   ransac은 기본적으로 일부의 데이터를 뽑아 그 데이터들로 임의의 정답을 찾고 다른 데이터에 얼마나 그 데이터가 들어 맞는지 확인하여, inlier를 찾는 과정임. 본 코드에서는 이미의 두선을 잡아서 그 교점을 vp를 두고 다른 모든 선들과 직선의 거리를 구했으며 일정 thr 미만으로 거리가 작을 때 inlier라고 판단하여 iter중에 가장 많은 inlier를 가진 경우를 gt라고 설정. 기본적으로 처음에 이미지에 z축 소실점으로 항하는 선이 가장 많을 것 이라고 가정  
   장면, 레이저, 실내, 빛이(가) 표시된 사진

   자동 생성된 설명
2. x,y축 소실점 찾기  
   x축 y축 성분은 단순 기울기를 구하고 atan을 구하여 각선의 각도를 구한후 -10~10 이면 y, 80~100 이면 x 성분으로 두었음, 각각을 1번 zvp를 찾았던 것처럼 똑같이 찾음  
   천장, 스크린샷, 실내, 그린이(가) 표시된 사진

   자동 생성된 설명천장, 실내, 바닥, 매장이(가) 표시된 사진

   자동 생성된 설명  
   천장, 실내, 그린, 빛이(가) 표시된 사진

   자동 생성된 설명천장, 실내, 바닥, 홀이(가) 표시된 사진

   자동 생성된 설명
3. 카메라 Scene Rectification  
   먼저 카메라 좌표계와 이미지 평면의 좌표를 동기화하는 calibration을 진행. 그후 소실점의 3개의 성분이 카메라의 좌표계와 동일해지는 카메라 위치를 찾고, 그곳에서 봤을 때 본래 이미지가 어떻게 보이는지 visualize  
   천장, 실내, 바닥이(가) 표시된 사진

   자동 생성된 설명천장, 스크린샷, 실내, 바닥이(가) 표시된 사진

   자동 생성된 설명
4. 임의의 카메라 포즈를 만들기  
   그냥 회전행렬은 sin cos등의 단순 가중합을 이용해서 보간하기는 힘듬, 쿼터니온은 회전을 일반 유클리드 공간의 벡터로 표현한 것인데 이는 가중 합이 가능. 회전행렬을 쿼터니온으로 바꾼후 가중 합으로 보간을 한 후에 다시 회전 행렬로 적용
5. Box를 만들고 만든 카메라 포즈를 rander  
   이미지 plane을 카메라 좌표계로 바꾸고, 박스 좌표와 이미지 좌표계를 주어진 카메라 회전, 이동행렬을 적용, K를 이용해 다시 image plane으로 각각을 보냄으로써 자연스러운 이미지들을 연출, 여기서 이미지의 검은줄은 랜더 과정에서 소숫점을 int로 옮길때의 오류. 이는 보간이나 image grid를 생성할때 더욱 촘촘히 생성하여 해결 가능하지만 시간 상 진행 불가  
   스크린샷, 그린, 라인, 천장이(가) 표시된 사진

   자동 생성된 설명레이저, 스크린샷, 빛, 라인이(가) 표시된 사진

   자동 생성된 설명  
   스크린샷, 라인, 그린, 레이저이(가) 표시된 사진

   자동 생성된 설명스크린샷, 그린, 레이저, 빛이(가) 표시된 사진

   자동 생성된 설명  
   5평면중 사이드 4평면은 카메라의 피치, 요 각도에 따라 랜더 순서가 바뀌어야 하지만, 그부분은 스킵함