

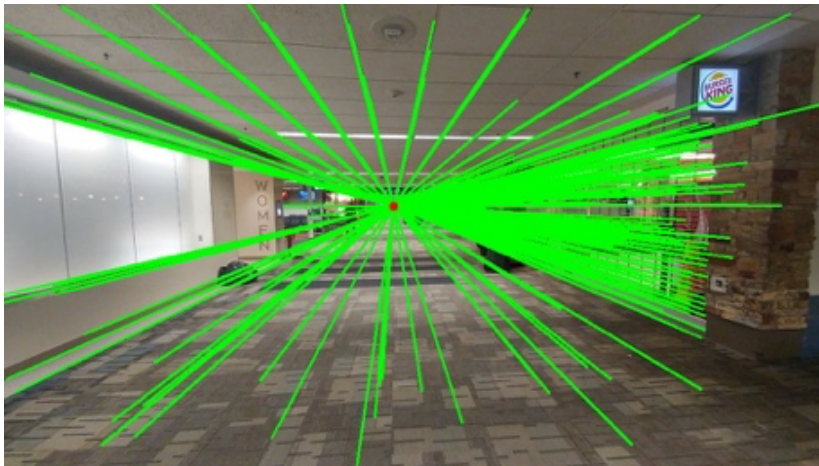
## AR/VR/XR assignment visualize 2

M2023067 김의찬

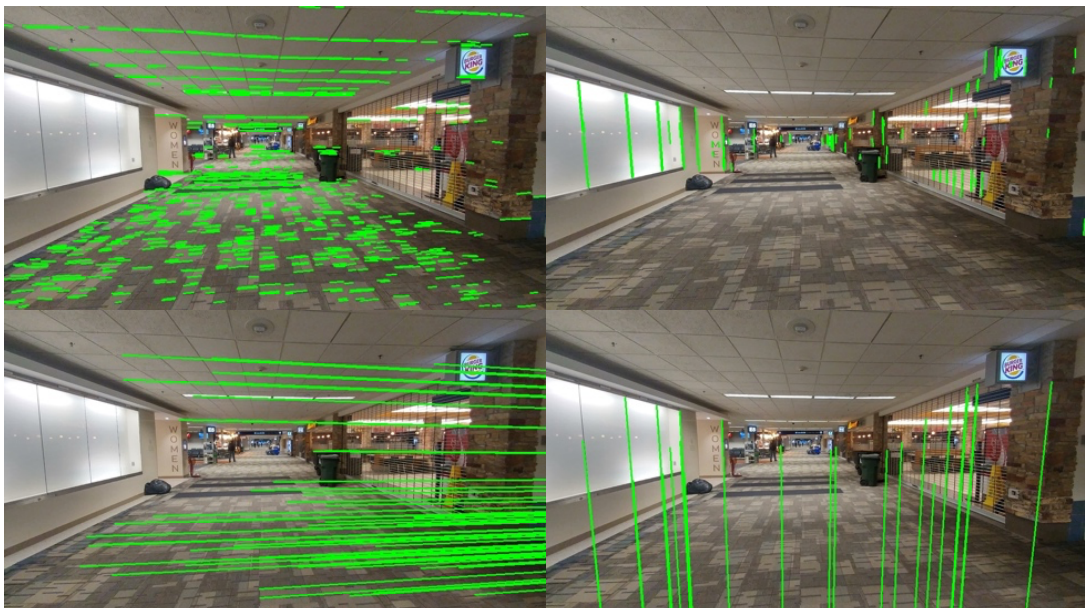
### 1. Find VP

find\_vp 함수는 RANSAC 알고리즘으로 소실점을 찾는 함수입니다. 2개의 무작위의 선을 고르고, 교점을 소실점이라 가정하고 나머지 선과 소실점과의 거리를 계산하고 일정 값 이하인 선의 수를 카운트 합니다. Iter수 만큼 반복합니다. 처음에는 z축 방향의 선이 많다고 가정하고 find vp 함수를 돌립니다. 이때 이미지 좌표계에서 바로 적용하는 것은 값이 크기 때문에 될 수 있습니다. 이를 방지하기 위해 먼저 임의의 Camera param K 를 설정하고, RANSAC을 진행합니다.

$$\begin{bmatrix} \tilde{u} \\ 1 \end{bmatrix} = \mathbf{K}^{-1} \begin{bmatrix} u \\ 1 \end{bmatrix}$$



x축 y축 소실점은 선의 수가 비슷할 수 있으므로 한번 clustering을 진행합니다. 기울기를 구하고 atan을 구하여 직선의 각도를 구한 후 -10~10 이면 y, 80~100 이면 x 성분으로 두었습니다. 각각을 find\_vp 함수를 통해 소실점을 찾아줍니다.

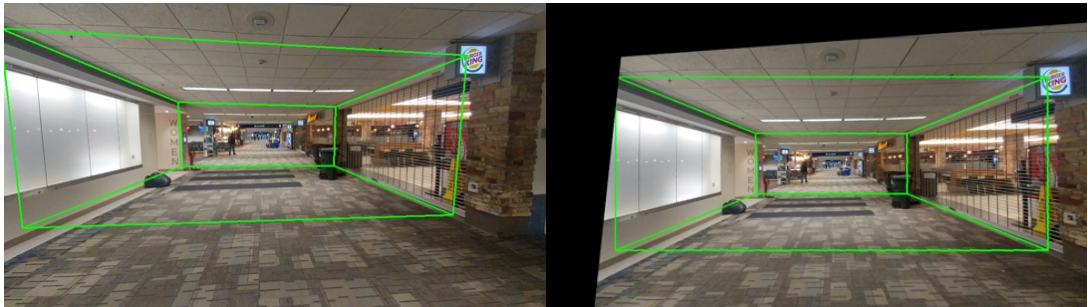
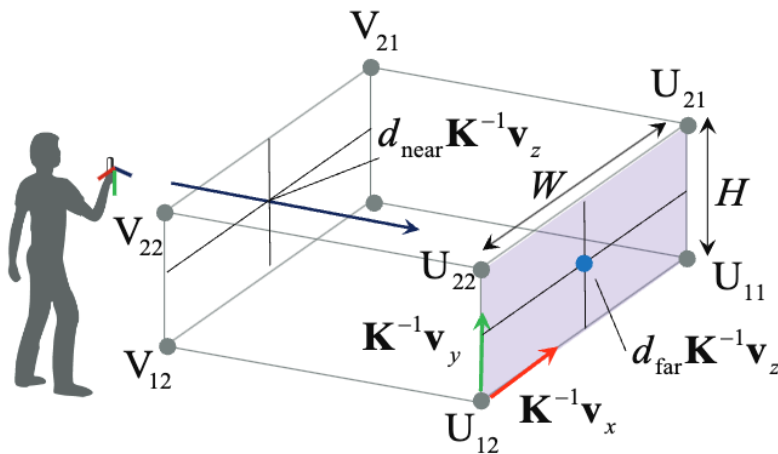


## 2. 카메라 Scene Rectification

먼저 카메라 좌표계와 이미지 평면의 좌표사이의 관계된 행렬을 찾는 calibration을 진행합니다. 이미지 plane의 중간으로부터  $f$ 만큼 떨어진 원점과 이미지의 3개의 소실점 좌표가 서로 수직이 되게하는 파라미터들을 찾습니다.

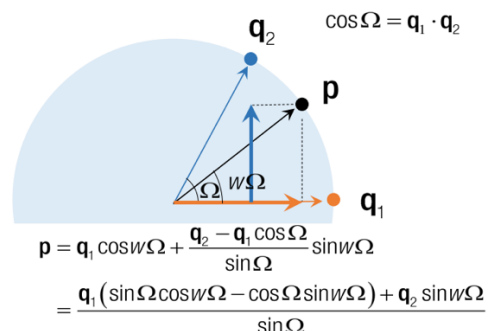
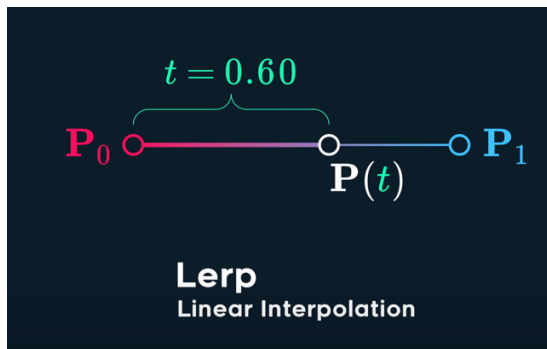
$$\mathbf{A} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \mathbf{0}, \quad p_x = -\frac{b_2}{b_1}, \quad p_y = -\frac{b_3}{b_1}, \quad f = \sqrt{\frac{b_4}{b_1} - (p_x^2 + p_y^2)}.$$

그 후 소실점의 3개의 성분이 카메라의 좌표계와 평행하게 하고, z소실점이 이미지 중앙에 오게 하는 행렬  $H$ 를 찾고, homography 행렬을 찾고 변환합니다. 카메라 좌표계에서 공간을 잘 표현하는 3D box 좌표들을 찾습니다.



## 3. 카메라 pose interpolation 함수 만들기

주요한 카메라 pose를 선언하고, 그 사이의 움직임을 부드럽게 하기 위해, interpolation을 진행합니다. Pose는 단순 lerp 함수를 이용해 interpolatoin하고, 회전행렬은 quaternions으로 변환 후 각도가 부드럽게 변하도록 계산하여 interpolation합니다.



#### 4. 3D rendering

제시된 주요 camera pose를 interpolation하고 RC 행렬들로 camera 좌표계를 변환하고, rendering하는 과정입니다. GetPlaneHomography 함수는 3D box중 평면을 이루는 3개의 좌표, 변환할 카메라 자세 행렬을 입력으로 받습니다. 먼저 3개의 좌표는 직사각형중 3점이기 때문에, 한 점을 먼저 찾습니다.

$$p_{22} = p_{21} + p_{12} - p_{11}$$

4점을 먼저 기존 이미지 plane 에 scale projection을 합니다.

$$\frac{K \times p}{|norm(K \times p)|}$$

그 후 카메라 포즈에 맞게 4점을 회전 이동을 진행합니다. 이 때 포인트들은 카메라 포인트들과 반대로 움직입니다.

$$tp = R^{-1} \times (p - C)$$

그 때의 이미지 plane에 scale projection을 시킵니다.

$$\frac{K \times tp}{|norm(K \times tp)|}$$

기존 이미지 plane의 n점과 새로운 image plane의 4점을 이용하여, 두 평면 사이의 관계인 Homography 행렬을 구합니다. 이를 통해 각각 5개의 평면을 구해서 새로운 이미지에 표현합니다.

