Computer Vision 기초에 대한 학습

Stereo Vision을 통한 Depth-map 형성

[2017년 수치해석 Term project 보고서]

전기공학과 201224138 배구환
전기공학과 201224184 홍태관
전기공학과 201524512 윤미린

1. Introduction

이번 텀 프로젝트의 주제인 “Stereo Vision”을 하게 된 동기는 저희 팀원들이 속해있는 학술동아리 프로젝트 경험에서 비롯되었습니다. 저희 동아리에서는 애완견로봇이나 자율주행모형차 등을 만든 경험이 있습니다. 기존 라인트레이서는 초음파 센서 등을 사용했고, 먼 거리의 장애물을 파악하지 못한다거나 차선을 파악하여 차량을 미리 제어하지 못한다는 한계가 있었습니다. 그래서 이 한계를 극복하기 위해 올해 만든 자율 주행 모형차에서는 비전 시스템을 새로이 도입했습니다. 이를 계기로 컴퓨터 비전에 대한 관심을 가졌고, “Stereo Vision”을 텀 프로젝트의 주제로 선정했습니다. Computer Vision에 대한 개념들을 학습하는데 있어서 전공과목으로 배운 Calculus, Signal Processing, Probability, Numerical Analysis 등을 사용했습니다. 이 과정에서 책과 강의실에 갇혀있는 수동적인 학습을 벗어나 우리가 관심을 가진 주제를 다루며, 학교에서 배운 지식들을 녹여낼 수 있는 좋은 주제였습니다.

Computer Vision이란 컴퓨터가 사람이 보는 것처럼 색상, 거리, 이동방향, 속도 등을 알 수 있게 만들어 주는 것입니다. 사람이 눈을 통해 들어온 시각 정보를 처리 하듯 컴퓨터는 카메라로 촬영한 이미지를 처리하여 정보를 얻습니다. Computer Vision을 활용하면, 컴퓨터를 사용하여 문자를 인식하는 OCR System을, 주변 환경 정보를 파악하여 Autonomous Driving System을, 멀티미디어에 사용하는 3D Graphics를 구축할 수 있습니다.

Stereo Vision은 Computer Vision의 한 분야로서, 사람이 두 눈의 시각차를 통해 물체의 원근감을 파악하는 것처럼 컴퓨터가 두 개의 카메라로 촬영한 이미지를 처리하여 피사체의 거리를 파악하는 것입니다. 이것을 이용하면 주변 장애물이 얼마나 떨어져 있는지 파악할 수 있습니다. 우리는 Stereo Vision을 사용하여 모바일 로봇, 자율주행 모형차 등에 적용할 수 있는 Depth Camera System을 만들어 보았습니다.

2. Method

1) 디지털 이미지

디지털 이미지는 픽셀로 이루어진 M × N 매트릭스입니다. 이미지를 설명하기 위하여 샘플링(sampling)과 양자화(quantization)이 무엇인지에 대해 먼저 설명하겠습니다. 이미지에서 샘플링이란 매트릭스를 이루는 픽셀들에 입력되는 빛의 강도를 전압 값으로 변환하여 저장하는 것입니다. 샘플링 주기가 짧을수록 픽셀의 개수가 많아지고 더 많은 정보를 입력 할 수 있어 높은 해상도를 가지게 됩니다.

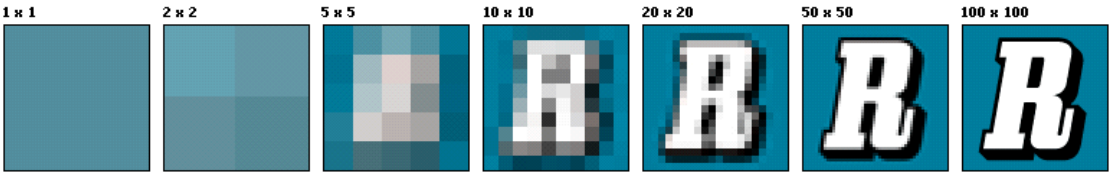


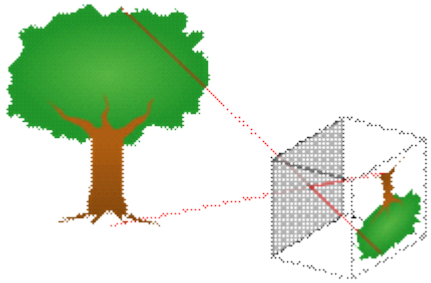
그림 1) 샘플링 주기에 따라 다른 해상도를 가지는 이미지

양자화는 저장된 전압 값을 가장 근접한 디지털 값으로 나타내 주는 것을 의미합니다. 이진 영상(binary image)은 학 픽셀 당 0과 1 즉 1bit로 이루어 진 영상입니다. LED가 켜져 있으면 1, 꺼져 있으면 0으로 나타내는 방식입니다. 흑백 영상(Gray-scale)의 경우 한 픽셀 당 0~255까지 8bit로 이미지의 명도만을 표시합니다. 마지막으로 색상 정보를 갖는 RGB 모델은 3원색을 요소로 저장하며 더 다양한 표현이 가능합니다. 예를 들어 True RGB의 경우는 한 픽셀 당 24bit의 정보를 표현하며, 24bit는 Red, Green, Blue 각 8bit씩 색상 정보를 나타냅니다. 우리가 읽는 일반적인 두께의 소설책을 텍스트로 저장하면 대략 250~300kb의 용량이 필요합니다. 이에 반해 이미지는 매우 큰 용량을 필요로 하는 데이터입니다. 4K 해상도의 사진을 예로 들어 간단히 계산해보면 3840x2160개의 각각의 픽셀에 24bit의 색상 정보를 표현하려면 3840x2160x24bit = 23.7MB의 용량이 필요합니다. 이미지는 매우 큰 용량을 갖는 데이터이며, 이를 처리해서 정보를 얻는 과정은 매우 높은 컴퓨터 자원을 필요로 합니다.

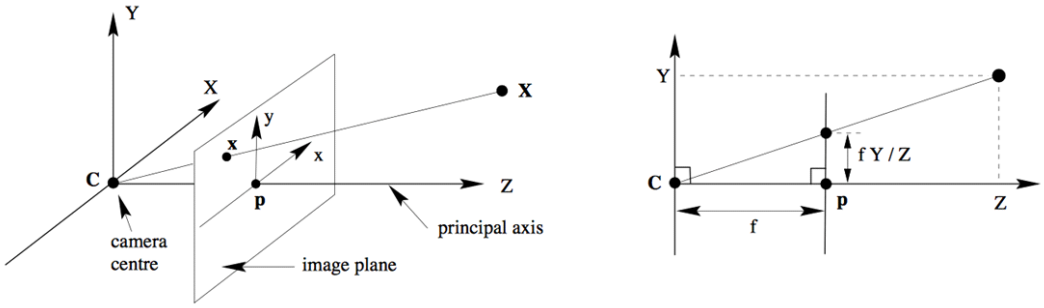
우리가 실제로 사용하는 스마트폰에서는 카메라의 하드웨어 성능보다 훨씬 좋은 화질의 이미지를 얻을 수 있습니다. 그 이유는 스마트폰에서 카메라로 촬영한 이미지의 픽셀 사이마다 보간(interpolation)해서 화질을 높이는 소프트웨어 보정 작업을 하기 때문입니다. 이 과정에서 스마트폰의 카메라는 여러 장의 이미지를 촬영하고 어떻게 픽셀 사이를 보간 할 것인지 판단하고 소프트웨어 보정을 통해 카메라 하드웨어의 한계 이상의 고해상도 이미지를 얻을 수 있습니다.

2) 이미지 형성

이미지가 맺히는 과정을 설명하기 위해 핀홀 모델을 설명 하겠습니다. 핀홀이란 말 그대로 핀으로 뚫은 구멍 같은 매우 작은 구멍을 말합니다. 이 작은 핀홀로 부터 빛을 받아들여 촬영하는 카메라를 핀홀 카메라라고 합니다.



핀홀 카메라의 원리는 물체에서 반사된 빛이 작은 구멍으로 들어와 이미지 평면에 맺히게 됩니다. 이 맺히는 과정을 수학적으로 모델링 한 것이 Pinhole camera model입니다.



<Pinhole camera model>

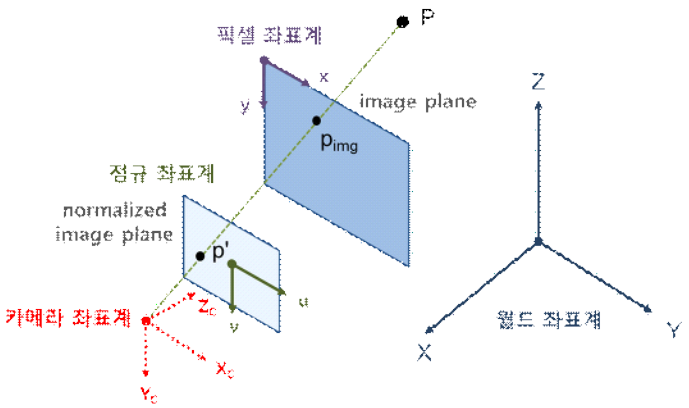
그림처럼 Pinhole camera model에서 간단한 수학 공식을 이용하면 상의 크기를 알 수 있습니다. 상의 크기를 x, 초점 거리를 f, 물체까지의 거리를 Z, 물체의 크기를 X라 두고 위의 그림에서 $x : f = X : Z$, $\therefore x = \frac{fX}{Z}$로 상의 크기를 구할 수 있습니다.

3) 이미지 기하학

이미지 기하학이란 3D 물체가 2D로 투영됐을 때, 특정한 면에 정확하게 matching 하는 과정을 뜻 합니다. 이를 설명하기 위해서는 일단 좌표계에 대해 설명하겠습니다..

이미지 기하학에서는 크게 월드좌표계, 카메라좌표계, 정규좌표계, 영상좌표계 4개가 있습니다.

월드 좌표계와 카메라 좌표계는 3차원 좌표계이고, 정규좌표계와 영상좌표계는 2D 좌표계입니다.



월드좌표계는 사물의 위치를 표현할 때 기준으로 삼는 좌표계로서 우리가 살고 있는 공간의 한 점을 기준으로 하는 좌표계라고 할 수 있습니다. 카메라 좌표계는 카메라를 기준으로 한 좌표계로 카메라 정면 광학축을 Z축, 카메라의 아래쪽을 Y축, 오른쪽을 X축으로 잡습니다.

영상좌표계는 우리가 실제로 보는 영상에 대한 좌표계로서 이미지 왼쪽 상단 모서리를 원점, 오른쪽 방향을 x축 증가방향 아래쪽을 y축 증가방향으로 합니다. 그리고 영상 좌표계의 x,y축에 의해 결정되는 평면을 이미지 평면이라고 합니다.

정규좌표계는 카메라 내부 파라미터의 영향을 제거한 가상의 좌표계입니다. 또한 정규화를 거친 좌표계이므로 단위가 없습니다. 카메라 초점과의 거리가 1인 가상의 이미지 평면을 정의하는 좌표계입니다. 원래의 이미지평면을 평행이동 시켜서 초점과의 거리를 1로 옮겨놓은 이미지 평면이라고 생각할 수 있습니다. 정규좌표계를 도입한 이유는 같은 사진을 찍더라도 카메라의 종류와 세팅에 따라 다른 영상이 얻어지므로 이로 인한 기하학적 해석의 불편함을 제거하기 위함입니다.

3차원 공간에 있는 점을 2차원 이미지평면에 투영시키는 변환을 투영변환이라 합니다.

앞의 핀홀 모델을 사용하여 비례식을 세워보면 쉽게 행렬을 구할 수 있는데 homogeneous(동차)좌표로 표현합니다.

월드좌표계 상의 한 점을 이미지평면 상의 한 점으로 변환시키는 행렬을 T(3x4행렬)라 하고, T를 분해하여 $T = KT_{pers}(1)[R|t]$로 표현할 수 있습니다. 이 때, T_pers(1)은 위에서 핀홀 모델링으로 구한 정규좌표계로의 (초점과의 거리가 1인) 투영변환을 말합니다. K는 카메라 내부 파라미터 행렬로 정규이미지 좌표를 픽셀 이미지 좌표로 바꾸어 준다. R|t 행렬은 이미지를 회전(rotation), 이동(translation)시키는 행렬입니다.

4) Homography(projective transformation)

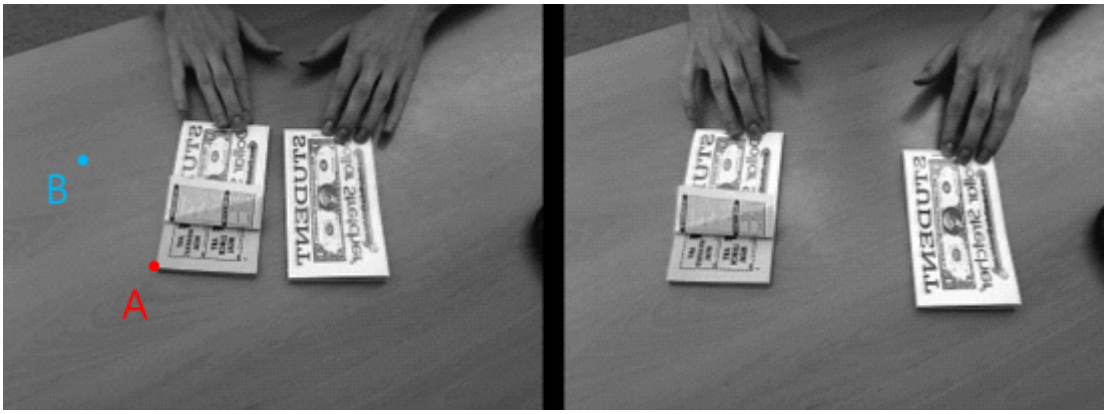
Planer surface 물체의 경우에는 3D 공간에서의 2D 이미지로의 임의의 원근투영변환을 두 이미지 사이의 homography로 모델링할 수 있습니다. 즉, 어떤 planer surface가 서로 다른 카메라 위치에 대하여 각각 다른 이미지 두 개로 투영되었을 때 그 이미지들의 관계를 homography로 표현할 수 있습니다. 따라서 가상의 시점에서 본 이미지로 변환을 하여 이미지를 정렬하거나, 반대로 이미지를 특정 영역으로 워핑(Warping)하여 원하는 이미지로 합성할 수 있습니다. 이후 Disparity를 계산하여 Depth-map을 구축할 때 정확한 거리 계산을 위해 두 카메라에서 촬영한 이미지의 시점을 가상의 plane으로 옮겨 Epipole line을 평행하게 만들어야합니다.

5) 카메라 캘리브레이션

실제 이미지는 사용된 렌즈, 렌즈와 이미지 센서와의 거리, 렌즈와 이미지 센서가 이루는 각 등 카메라 내부의 기구적인 부분에 의해서 크게 영향을 받습니다. 따라서, 3차원 점들이 영상에 투영된 위치를 구하거나 역으로 영상좌표로부터 3차원 공간좌표를 복원할 때에는 이러한 내부 요인을 제거해야만 정확한 계산이 가능해집니다. 그리고 이러한 내부 요인의 파라미터 값을 구하는 과정을 카메라 캘리브레이션이라 부릅니다. 카메라는 각각 고유의 내부 파라미터를 가지며, 이미 알고있는 체스판을 여러 각도와 여러 거리에서 촬영한 샘플로 캘리브레이션 작업을 수행합니다. 정확한 패러미터를 측정하기 위해서는 각기 다른 각도와 거리에서 촬영한 샘플이 필요합니다.

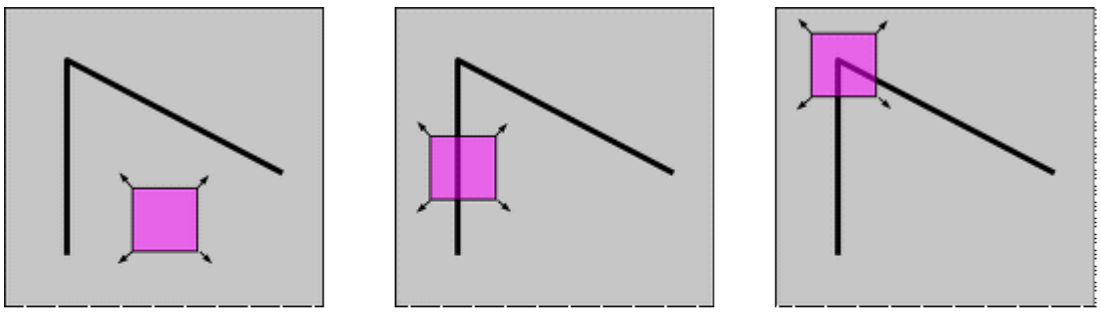
6) 특징점

스테레오 이미지에서 거리 정보를 정확하게 얻기 위해서는 두 이미지를 잘 매칭시켜야 합니다. 그러기 위해서는 실제로 같은 점을 정확하게 찾아내서 매칭시켜야 합니다. 이미지는 매우 큰 데이터이기 때문에 모든 점들을 비교하는 것은 매우 오래 걸리는 작업이며 컴퓨터 자원을 비효율적으로 쓰는 것입니다. 그렇기 때문에 우리는 특징점이라는 것을 찾아 매칭을 시킵니다. 이 특징점은 주변과 명확하게 구분되는 점이어야 합니다.



위 그림에서 첫 번째 사진과 두 번째 사진의 같은 점을 찾을 때 A점은 사물의 모서리 부분으로 주변과 구분이 잘 되어 특징점으로 적합하지만, B점은 배경의 한 부분으로 주변과 구분이 되지 않아 특징점으로 부족합니다. A점과 같이 좋은 특징점이 되기 위한 조건은 첫 째, 물체의 형태나 크기, 위치가 변해도 쉽게 식별이 가능한 점, 둘째, 카메라의 시점, 조명이 변해도 영상에서 해당 지점을 쉽게 찾아낼 수 있는 점입니다. 이러한 조건들을 제일 잘 만족시키는 것이 바로 코너점(corner point)입니다.

특징점을 검출하는 방법은 매우 많지만, 보고서에서는 코너점을 찾는 가장 대표적인 방법인 Harris corner detector만 소개 하겠습니다. 영상에서 코너를 찾는 기본적인 아이디어는 아래 그림과 같이 영상에서 작은 윈도우를 조금씩 이동시킬 때, 코너점의 경우는 모든 방향으로 영상변화가 커야하는 점입니다.



Harris corner detector는 1980년 Moravec의 아이디어를 수정 보완한 것으로서, 이 방법을 설명하기 위해 수식을 사용겠습니다. 먼저 $(\Delta x, \Delta y)$만큼 윈도우를 이동 시킬때 영상의 SSD(sum of squared difference) 변화량 E는 다음과 같습니다.

$$E(\Delta x,\Delta y)=\sum_{W}\left[I(x_i+\Delta x,y_i+\Delta y)-I(x_i,y_i)\right]^2$$

이 때, shift값 $(\Delta x, \Delta y)$이 매우 작다고 가정하고 그레디언트(gradient)를 이용하여 I를 선형 근사하면(1차 테일러 근사),

$$I(x_i+\Delta x, y_i+\Delta y) \approx I(x_i,y_i)+\begin{bmatrix} I_x(x_i,y_i) & I_y(x_i,y_i) \end{bmatrix}\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

$$E(\Delta x,\Delta y) = \sum_{W} \left[I(x_i+\Delta x, y_i+\Delta y) - I(x_i,y_i) \right]^2$$

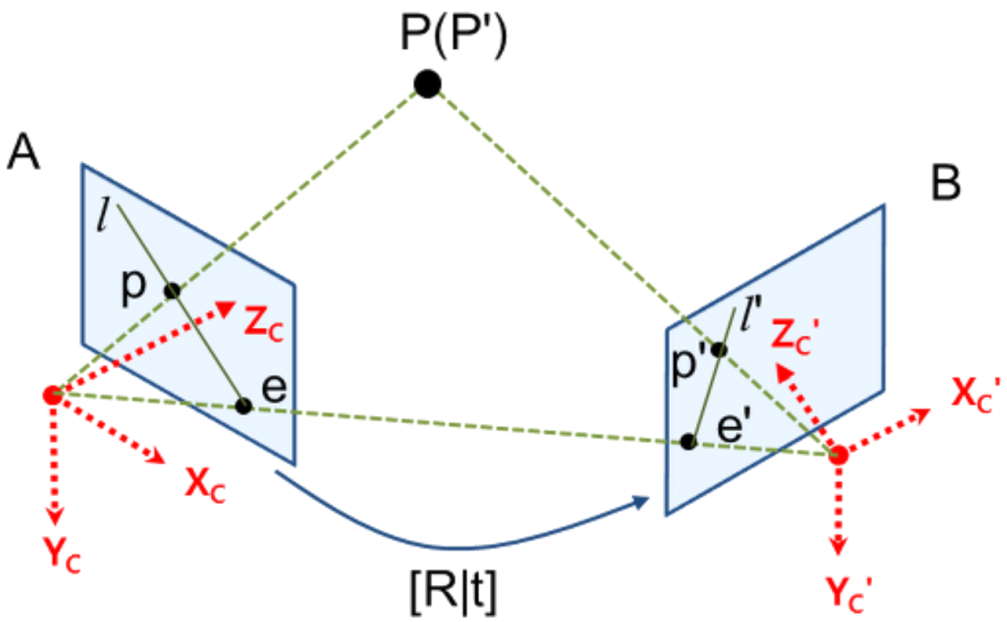
$$\approx \sum_{W} \left[I(x_i,y_i)+\begin{bmatrix} I_x(x_i,y_i) & I_y(x_i,y_i) \end{bmatrix}\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} - I(x_i,y_i) \right]^2$$

$$= \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} \begin{bmatrix} \sum_{W} I_x(x_i,y_i)^2 & \sum_{W} I_x(x_i,y_i) I_y(x_i,y_i) \\ \sum_{W} I_x(x_i,y_i) I_y(x_i,y_i) & \sum_{W} I_y(x_i,y_i)^2 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

$$= \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} M \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

가 됩니다. 이 때 2×2 행렬 M의 두 고윳값(eigenvalue)를 $\lambda_1, \lambda_2 (\lambda_1 \geq \lambda_2)$라 하면 영상 변화량 E는 윈도우를 λ_1의 고유벡터(eigenvector) 방향으로 이동 시킬 때 최대가 되고, λ_2의 고유벡터 방향으로 이동 시킬 때 최소가 됩니다. 또한 두 고유값 λ_1, λ_2는 해당 고유벡터 방향으로의 실제 영상 변화량(E) 값이 됩니다. 따라서, M의 두 고유값 λ_1, λ_2를 구했을 때. 두 값이 모두 큰 값이면 'corner point', 모두 작은 값이면 'flat area', 하나는 크고 하나는 작은 값이면 'edge' 영역으로 판단 할 수 있습니다. Harris corner detector의 특징을 살펴보면, 영상의 평행이동, 회전변화에는 불변이고 아핀 변화, 조명 변화에도 어느 정도는 강인성을 가지고 있습니다. 하지만 영상의 크기(scale)의 변화에는 영향을 받기 때문에 응용에 따라서는 여러 영상 스케일에서 특징점을 뽑을 필요가 있습니다.

7) 에피폴라 기하학



위 그림과 같이 3차원 공간상의 한 점 P가 영상 A에서는 p에 투영되고, 영상 B에서는 p'에 투영됐다고 하겠습니다. 이 때, 두 카메라 원점을 잇는 선과 이미지 평면이 만나는 점 e, e'을 epipole이라 부르고 투영점과 epipole을 잇는 직선 l, l'을 epiline (또는 epipolar line)이라 부릅니다. Epiline은 3차원의 점 P와 두 카메라 원점을 잇는 평면(epipolar plane)과 이미지 평면과의 교선으로도 볼 수 있습니다.

두 카메라 위치 사이의 기하학적 관계 [R|t]를 알고 있고 영상 A에서의 영상좌표 p를 알고 있을 때, 영상 B에서 대응되는 점 p'의 좌표를 구하는 문제를 생각해 보겠습니다. 이 때, 점 P까지의 거리(depth) 정보를 모른다면 영상좌표 p로부터 투영되기 전의 3차원 좌표 P를 복원할 수는 없습니다. 따라서 점 P가 영상 B에 투영된 좌표 p' 또한 유일하게 결정할 수 없습니다. 하지만 점 P는 A 카메라의 원점과 p를 잇는 직선(ray) 상에 존재하기 때문에 이 직선을 영상 B에 투영시키면 점 p'이 이 투영된 직선 위에 있음은 알 수 있습니다. 이 투영된 직선이 바로 epiline l' 입니다.

정리하면, 'A의 영상좌표 p로부터 대응되는 B의 영상좌표 p'을 유일하게 결정할 수는 없지만 p'이 지나는 직선인 epiline l'은 유일하게 결정할수 있다' 입니다. 그리고 한 영상좌표로부터 다른 영상에서의 대응되는 epiline을 계산해주는 변환행렬이 Fundamental Matrix, Essential Matrix 입니다. 즉, 서로 다른 두 시점에서 찍은 영상좌표들 사이에는 Fundamental Matrix, Essential Matrix를 매개로 하는 어떤 변환 관계가 성립하는데, Epipolar Geometry에서는 이 변환관계를 바탕으로 여러 기하학적 문제를 풀게 됩니다.

물체가 카메라에 가까울수록 Disparity가 커지고, 멀수록 줄어듭니다. 만약 물체가 카메라로부터 아주 멀리 떨어져 있다면 Disparity는 0에 가까울 것입니다. 카메라의 파라미터를 안다면 픽셀별로 유효한 관측 거리를 추정할 수 있습니다. 우리가 가진 웹 카메라의 패러미터로 계산해 본다면, Disparity가 10픽셀만큼 부정확해도 물체와의 추정 거리는 5m 이상의 오차를 갖게 될 것입니다. Stereo Vision system은 현재 Mobile Robot과 Autonomous Vehicle이 주변의 환경 정보를 파악하는데 쓰이고 있습니다. 만약 거리 측정을 부정확하게 하는 경우 탑승자나 보행자가 위험에 처하는 심각한 문제에 처할 수 있습니다.

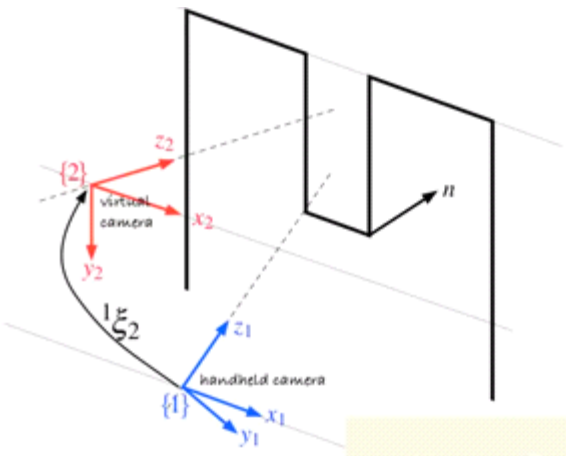
3. Result

1) 시점 사이의 호모그래피를 구하고 이미지를 변환하기

[Image Alignment]

먼저 이미지의 시점을 바꿔 정면에서 바라본 이미지로 변환하는 예제를 살펴보겠습니다. 샘플 이미지는 부산대학교 본관을 건물 밑에서 위로 바라본 시점으로 촬영한 이미지입니다.

이 이미지를 공중에서 바라본 시점으로 변환하는 호모그래픽 관계를 구해서, 건물의 외곽선이 직사각형으로 나오도록 보정해 보겠습니다.



$$H_E \simeq R + \frac{t}{d} n^T$$

첫 번째 접근방법으로는 카메라를 들고 있는 handled camera coordinate와 virtual camera coordinate 사이의 관계를 안다면 두 시점 사이의 호모그래픽 관계를 구할 수 있습니다.

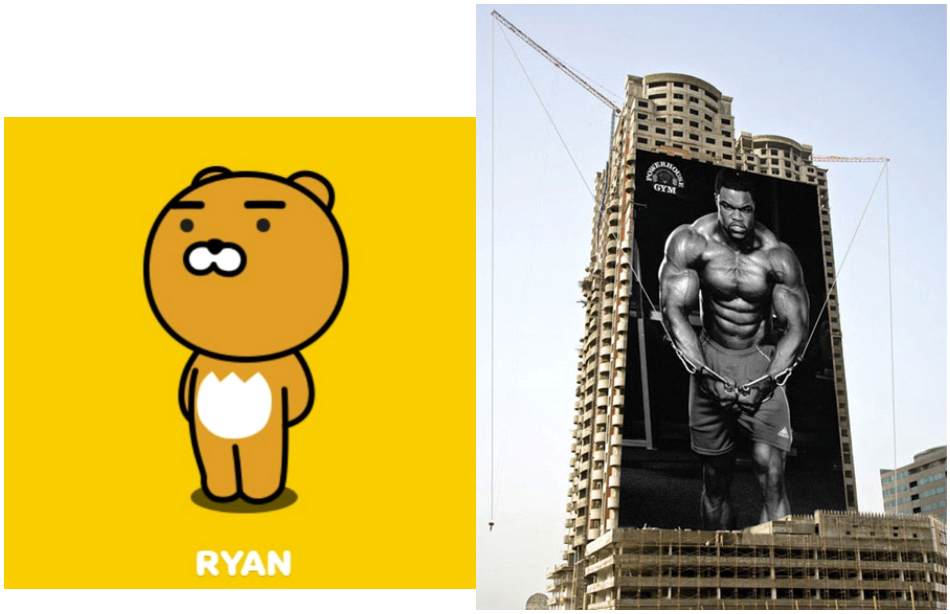
두 번째 방법으로는 원본이미지의 코너점 4개와 변환할 시점의 이미지 코너 4개를 입력받아 두 이미지 간의 호모그래픽 관계를 구하는 것입니다.

$$p^2 = H^{12}p^1$$

호모그래픽 관계를 구한다면 모든 픽셀간의 변환을 구할 수 있습니다.

변환된 이미지는 마치 건물의 중앙에서 바라본 것과 같은 시점을 보입니다.

[Virtual Ads on Building]
반대로 이미지 원본을 갖고 있고, 원하는 사각형으로 원근 변환을 해보겠습니다. 샘플 이미지로는 캐릭터의 이미지를 사용하여 건물 외벽의 옥외 광고 부분으로 캐릭터의 이미지를 변환해 보겠습니다.

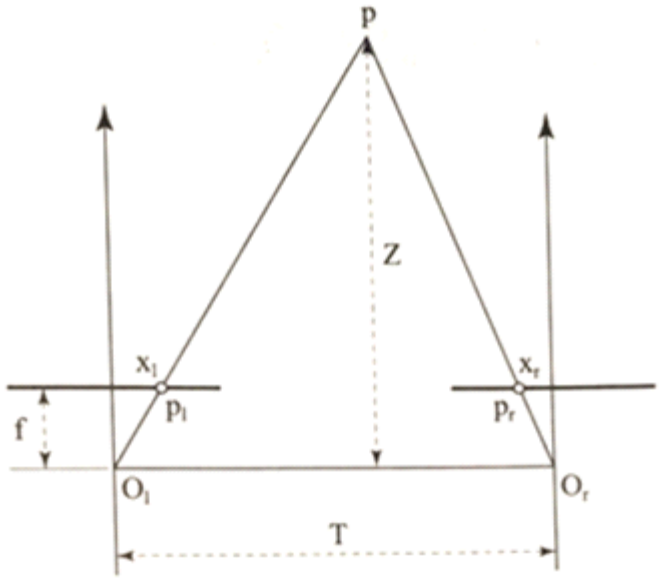


작성한 프로그램은 사용자가 마우스로 각 코너점을 지정해 줄 수 있습니다. 호모그래픽 매트릭스는 가역적이기 때문에 옥외광고물의 광고 사진을 직사각형 형태의 이미지로 보정할 수 있고, 샘플 이미지로 사용한 캐릭터 사진을 그 자리로 변환시킬수도 있습니다.

캐릭터의 사진이 옥외광고의 부분으로 변환되었습니다. 반대로 옥외광고물의 사진에 호모그래픽 매트릭스의 역행렬 $inv(H)$ 를 이용하면 직사각형의 이미지로 보정할 수 있습니다.

2) 스테레오 비전 형성

두 이미지가 캘리브레이션 되어있고, 에피폴 라인이 평행이 되도록 보정되었다면, 매칭 포인트들로부터 거리정보(Depth-map)를 환산할 수 있습니다. 즉 이미지 상의 매칭 포인트의 픽셀 좌표와 실제 물체(Object)까지의 거리(Disparity)를 환산할 수 있습니다. 아래는 스테레오 기하 방정식을 보여주고 있습니다. 따라서 두 개의 캘리브레이션된 카메라로 촬영한 이미지로부터 물체까지의 거리를 알아내는 문제를 다시 공식화 할 수 있습니다. 이 경우에는 Epipole line이 두 이미지 간의 모든 매칭 포인트들에 대해 수평이 되도록 이미지를 교정할 수 있습니다. 그러면 그 Disparity는 왼쪽 카메라 이미지와 비교해 오른쪽 카메라 이미지의 특정 지점이 움직인 픽셀의 수로 정의됩니다.



우리 팀은 두 개의 카메라로 스테레오 이미지를 촬영하고, 이로부터 실시간 Depth-map을 만들어내는 프로그램을 만들지는 못했습니다. 다만 캘리브레이션 된 두 이미지를 통해 Depth-map이 어떻게 형성되는지 확인할 수 있었습니다.