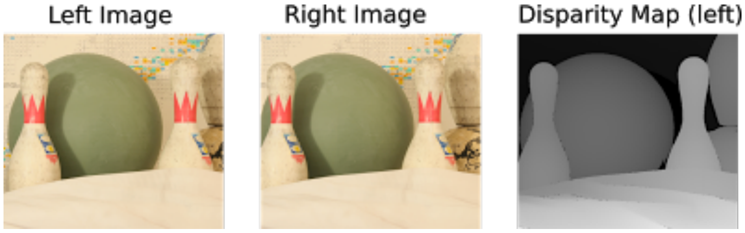
Stereovision looks at two rectified images to gain information about the 3D position of different features in the image. Rectified images mean the images are taken along the same plane. So, an image taken from a left camera may have a point the pixel location x, y. In its paired rectified image, that same feature will appear in x’, y—the feature will be in the same row but has moved to a different pixel horizontally. The more this feature has horizontally moved, the closer it is the camera. We see this phenomenon with human vision, a still finger held close to our eyes moves a lot if we close the left eye then switch to closing the right, and when it’s further it appears to move less with the same process.

How much features moves in pixels from one camera’s image to the other camera’s image is known as *disparity*. Advanced algorithms have descriptors for features in one image, and try and find them in the other. They will be in the same row when the pictures are rectified. The disparity image is basically a depth map—each pixel is an encoding for the distance moved for that pixel in the original images. Darker colors mean the feature at the pixel moved very little between the images, while lighter colors mean it moved a lot and must be closer.



(https://sites.google.com/site/5kk73gpu2010/_/rsrc/1289558676808/assignments/stereo-vision/overview.png?height=121&width=400)

Linear algebra can be used with stereovision to convert two dimensional coordinates in an image to a three dimensional point in real space. We want to create a matrix that by multiplied by a points’ coordinates on an image to find its position in 3D space. This matrix is known as the projection matrix, P.

$$P = K * [\, r \,| t\,]$$

where P is the projection matrix, K is the intrinsic matrix, and [r|t] is the extrinsic matrix, made up of the rotation matrix, r, and the translation vector, t. The intrinsic matrix contains parameters inherent to the camera, like its focal length, while the extrinsic matrix refers to the pose and rotation of the camera when the image was taken.

The intrinsic matrix K is made up of three matrices that are encoded with how the camera will translate, scale and shear a point in 3D space to a coordinate on the image.

$$\text{K} = \begin{matrix} 1 & 0 & X_0 \\ 0 & 1 & Y_0 \\ 0 & 0 & 1 \end{matrix} \quad \text{x} \quad \begin{matrix} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 0 & 1 \end{matrix} \quad \text{x} \quad \begin{matrix} 1 & s/f_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix} \quad = \quad \begin{matrix} f_x & s & X_0 \\ 0 & f_y & Y_0 \\ 0 & 0 & 1 \end{matrix}$$

where X_0 is the x value of the principle point, and Y_0 is the y value of the principle point, s is the axis skew, and f_x and f_y are the x and y values of the focal length, respectively.

This matrix fits in with the rotation matrix and translation vector to a bigger equation. A 3D coordinate (X,Y,Z) is multiplied on the righthand side of the equation by the extrinsic and intrinsic matrix. It equals the new coordinate in the image (u,v) multiplied by the scaling factor.

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

(http://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html)

For stereovision, we are considering this equation twice for each point, once for the left image and once for the right image. Both images have the same X Y Z vector, because they are dealing with the same point in 3D space. To compute a point in 3D space from a pair of rectified images, we consider a disparity map. With no rotation and just translation in the x direction (the baseline) as there is with rectified images, the equations become simpler. To solve for the X Y Z coordinates of a pixel in an image, we consider some of the camera's features, and the disparity encoded for that pixel in the disparity map image.

$$X = \frac{(u - X_0) *\ b}{d}$$

$$Y = \frac{(v - Y_0) *\ b}{d}$$

$$Z = \frac{f *\ b}{d}$$

where u is the row of the pixel, v is the column of the pixel, X_0 and Y_0 are the x and y dimensions of the principle point, respectively, b is the baseline and f is the focal length $\sqrt{{f_x}^2 + {f_y}^2}$.