Q2.1.

(1) Roll : $\Omega = \Omega_z$

$$\begin{bmatrix} X(t) \\ Y(t) \\ Z(t) \end{bmatrix} = \begin{bmatrix} \cos(\Omega t) X_0 - \sin(\Omega t) Y_0 \\ \sin(\Omega t) X_0 + \cos(\Omega t) Y_0 \\ Z_0 \end{bmatrix}$$

$$(x(t), y(t)) = \left(\frac{X(t)}{Z(t)}, \frac{Y(t)}{Z(t)} \right) f$$

$$= f \left(\frac{\cos(\Omega t) X_0 - \sin(\Omega t) Y_0}{Z_0}, \frac{\sin(\Omega t) X_0 + \cos(\Omega t) Y_0}{Z_0} \right)$$

$$\frac{d}{dt} x(t) = \frac{-\Omega \sin(\Omega t) X_0 - \Omega \cos(\Omega t) Y_0}{Z_0} \cdot f$$

$$\frac{d}{dt} y(t) = \frac{-\Omega \sin(\Omega t) Y_0 + \Omega \cos(\Omega t) X_0}{Z_0} \cdot f$$

$$(v_x, v_y) = \frac{d}{dt} (x(t), y(t)) \Big|_{t=0}$$

$$= \left(\frac{-\Omega Y_0 f}{Z_0}, \frac{\Omega X_0 f}{Z_0} \right)$$

$$= \Omega \left(-\frac{Y_0}{Z_0} f, \frac{X_0}{Z_0} f \right) = \Omega_z (-y, x)$$

(2) Tilt : $\Omega = -\Omega_x$

$$\begin{bmatrix} X(t) \\ Y(t) \\ Z(t) \end{bmatrix} = \begin{bmatrix} X_0 \\ \cos(\Omega t) Y_0 - \sin(\Omega t) Z_0 \\ \sin(\Omega t) Y_0 + \cos(\Omega t) Z_0 \end{bmatrix}$$

$$(x(t), y(t)) = \left(\frac{X(t)}{Z(t)}, \frac{Y(t)}{Z(t)} \right) f$$

$$= f \left(\frac{X_0}{\sin(\Omega t) Y_0 + \cos(\Omega t) Z_0}, \frac{\cos(\Omega t) Y_0 - \sin(\Omega t) Z_0}{\sin(\Omega t) Y_0 + \cos(\Omega t) Z_0} \right)$$

$$\frac{d}{dt}x(t) = \frac{X_0\left(\Omega\cos(\Omega t)Y_0 - \Omega\sin(\Omega t)Z_0\right)}{-\left(\sin(\Omega t)Y_0 + \cos(\Omega t)Z_0\right)^2} \cdot f$$

$$\frac{d}{dt}y(t) = \frac{\left(-\Omega\sin(\Omega t)Y_0 - \Omega\cos(\Omega t)Z_0\right)\left(\sin(\Omega t)Y_0 + \cos(\Omega t)Z_0\right) - \left(\cos(\Omega t)Y_0 - \sin(\Omega t)Z_0\right)\left(\Omega\cos(\Omega t)Y_0 - \Omega\sin(\Omega t)Z_0\right)}{\left(\sin(\Omega t)Y_0 + \cos(\Omega t)Z_0\right)^2} \cdot f$$

$$(v_x, v_y) = \frac{d}{dt}\left(x(t), y(t)\right)\Big|_{t=0}$$

$$= \left(\frac{\Omega Y_0 X_0}{-Z_0^2}\cdot f,\ \frac{-\Omega Z_0^2 - \Omega Y_0^2}{Z_0^2}\cdot f\right)$$

$$= -\Omega\left(\frac{X_0 f}{Z_0}\cdot\frac{Y_0 f}{Z_0}\cdot\frac{1}{f},\ \left(1 + \frac{Y_0^2 f^2}{Z_0^2}\cdot\frac{1}{f^2}\right)f\right)$$

$$= \Omega \times \left(\frac{xy}{f},\ f\left(1 + \frac{y^2}{f^2}\right)\right)$$

(3) Pan : $\Omega = \Omega_Y$

$$\begin{bmatrix} X(t) \\ Y(t) \\ Z(t) \end{bmatrix} = \begin{bmatrix} \cos(\Omega t)X_0 + \sin(\Omega t)Z_0 \\ Y_0 \\ -\sin(\Omega t)X_0 + \cos(\Omega t)Z_0 \end{bmatrix}$$

$$(x(t), y(t)) = \left(\frac{X(t)}{Z(t)},\ \frac{Y(t)}{Z(t)}\right)f$$

$$= f\left(\frac{\cos(\Omega t)X_0 + \sin(\Omega t)Z_0}{-\sin(\Omega t)X_0 + \cos(\Omega t)Z_0},\ \frac{Y_0}{-\sin(\Omega t)X_0 + \cos(\Omega t)Z_0}\right)$$

$$\frac{d}{dt}x(t) = \frac{\left(-\Omega\sin(\Omega t)X_0 + \Omega\cos(\Omega t)Z_0\right)\left(-\sin(\Omega t)X_0 + \cos(\Omega t)Z_0\right) - \left(\cos(\Omega t)X_0 + \sin(\Omega t)Z_0\right)\left(-\Omega\cos(\Omega t)X_0 - \Omega\sin(\Omega t)Z_0\right)}{\left(-\sin(\Omega t)X_0 + \cos(\Omega t)Z_0\right)^2} \cdot f$$

$$\frac{d}{dt}\,y(t) = \frac{Y_0\left(\Omega\cos(\Omega t)X_0 + \Omega\sin(\Omega t)Z_0\right)}{\left(-\sin(\Omega t)X_0 + \cos(\Omega t)Z_0\right)^2}\cdot f$$

$$(v_x, v_y) = \frac{d}{dt}\left(x(t),\ y(t)\right)\Big|_{t=0}$$

$$= \left(\frac{\Omega Z_0^2 + \Omega X_0^2}{Z_0^2}\cdot f\ ,\ \frac{\Omega Y_0 X_0}{Z_0^2}\cdot f\right)$$

$$= \Omega\left(f\left(1+\frac{X_0^2 f^2}{Z_0^2}\cdot\frac{1}{f^2}\right),\ \frac{X_0 f}{Z_0}\cdot\frac{Y_0 f}{Z_0}\cdot\frac{1}{f}\right)$$

$$= \Omega_Y\left(f\left(1+\frac{x^2}{f^2}\right),\ \frac{xy}{f}\right)$$

Q2.2

Let extrinsic_matrix = A (3×4)

intrinsic-matrix = K (3×3)

$$\begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = K \cdot A \cdot \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

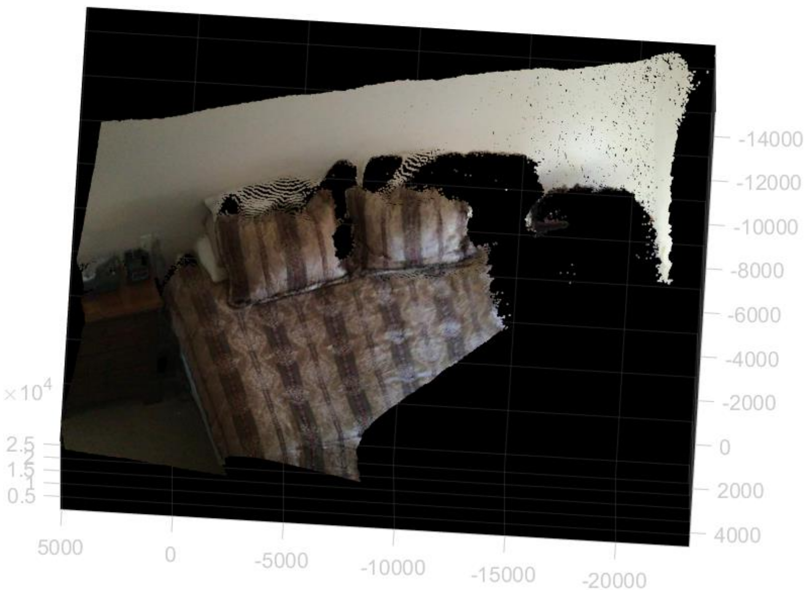
and we are given pixel index $\begin{bmatrix} x \\ y \end{bmatrix}$, K and A.

First, w = depthImage (x,y), then we get pixel position $\begin{bmatrix} wx \\ wy \\ w \end{bmatrix}$ in homogeneous coordinates.

Second, do $\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = K^{-1} \begin{bmatrix} wx \\ wy \\ w \end{bmatrix}$ we get the camera coordinates

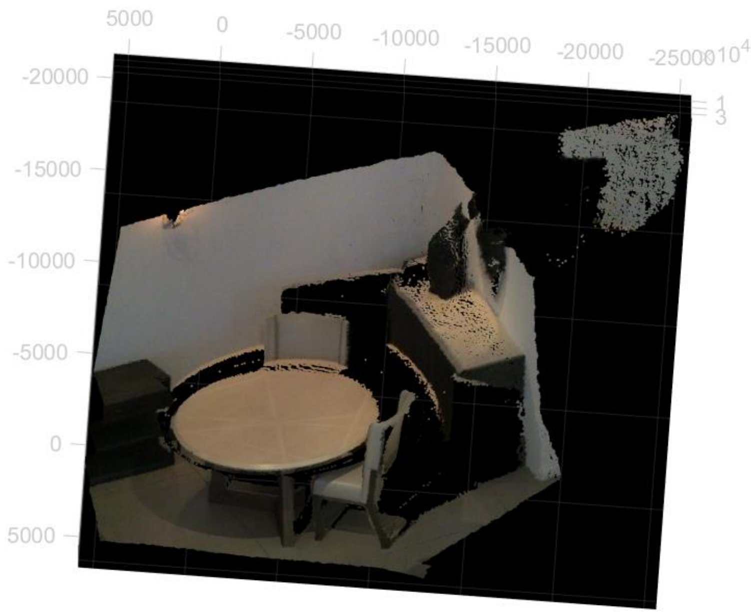
Third, do $\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} & A & \\ 0 & 0 \; 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}$ we get the world coordinates

The 3D visualization of point clouds for the three images are shown below:



Dataset_1

Dataset_2



Dataset_3

Q2.3

In this question we do some rotations and translations. The 3 rotations --- "pan", "roll", "tilt" should apply to the camera coordinates, while the translation should apply to the world coordinates. Even though we are translating the camera, but in the real calculation we need to apply the translation matrix in the world coordinates because (Tx, Ty, Tz) are given with respect to the X, Y, Z-axis.

I first calculated the new extrinsic matrix, and applied it to the original 3D points in world coordinates.

Original extrinsic matrix is $R[I|-C_w]$

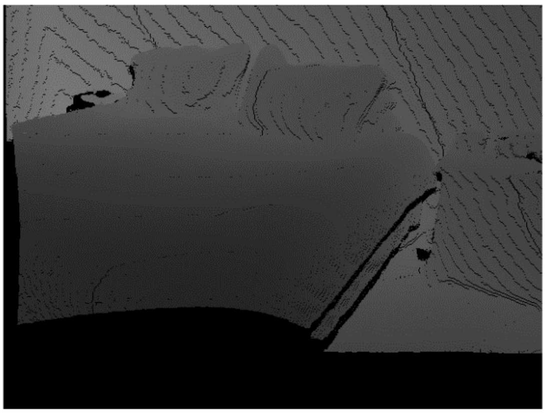
$$C_w' = C_w + \begin{bmatrix} T_x \\ T_Y \\ T_Z \end{bmatrix} t = C_w + T$$

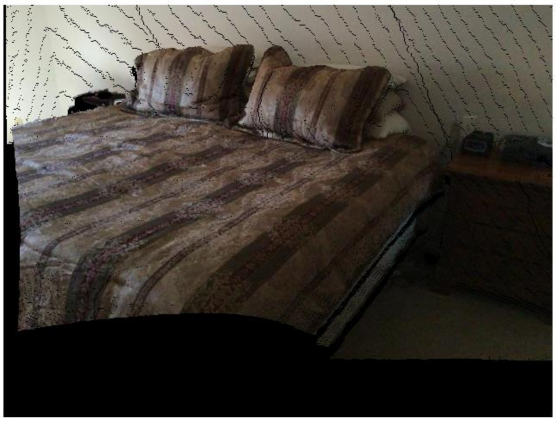
Thus the new extrinsic matrix is

$$R[I|-C_w'] = R[I|-C_w-T] = [R \quad -RC_w-RT]$$

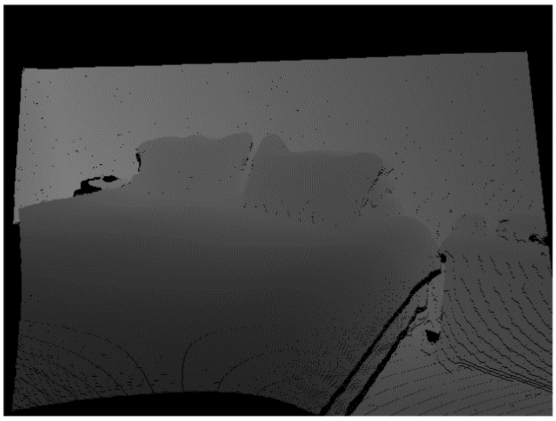
Second, I defined rotation matrix (w.r.t camera coordinates) in the function *rotation*, which output the rotated camera coordinates.

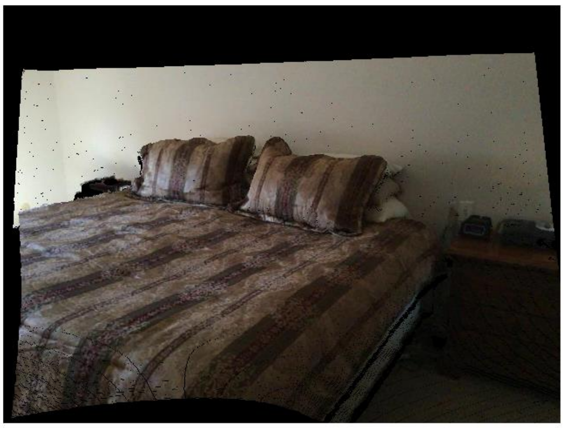
Below are some example results.
Results where omegaT=0, translationVector = (2000,0,0) :





Results where omegaT=pi/20, translationVector = (2000,0,0) :





I noticed that there were some black lines and spots on the projected images, especially on the part around the edges of the images. I think there are two reasons. First, in the origin depth image there are some zero values, which means that the corresponding 3D points are on the projection plane. So when we look at pixel (wx, wy, w) at the position where the value of the depth image is zero, we will lose the corresponding 3D point as $w = 0$. Second, when we do projection, we round the results to get integers for pixel positions.

Q2.4

For each data set I generate 6 movies, namely a grey image movie and RGB image movie for each of the 3 rotation and translation pairs, XZ, YX and ZY. For each movie, omega is equal to pi/60, step size for the translation is 500 and there are 20 iterations.
See the results in the folder *movies*