



RGBD Cameras' Per-Pixel Calibration and Natural 3D Reconstruction on GPU

Dr. Lau, advisor

Dr. Hastings, committee member

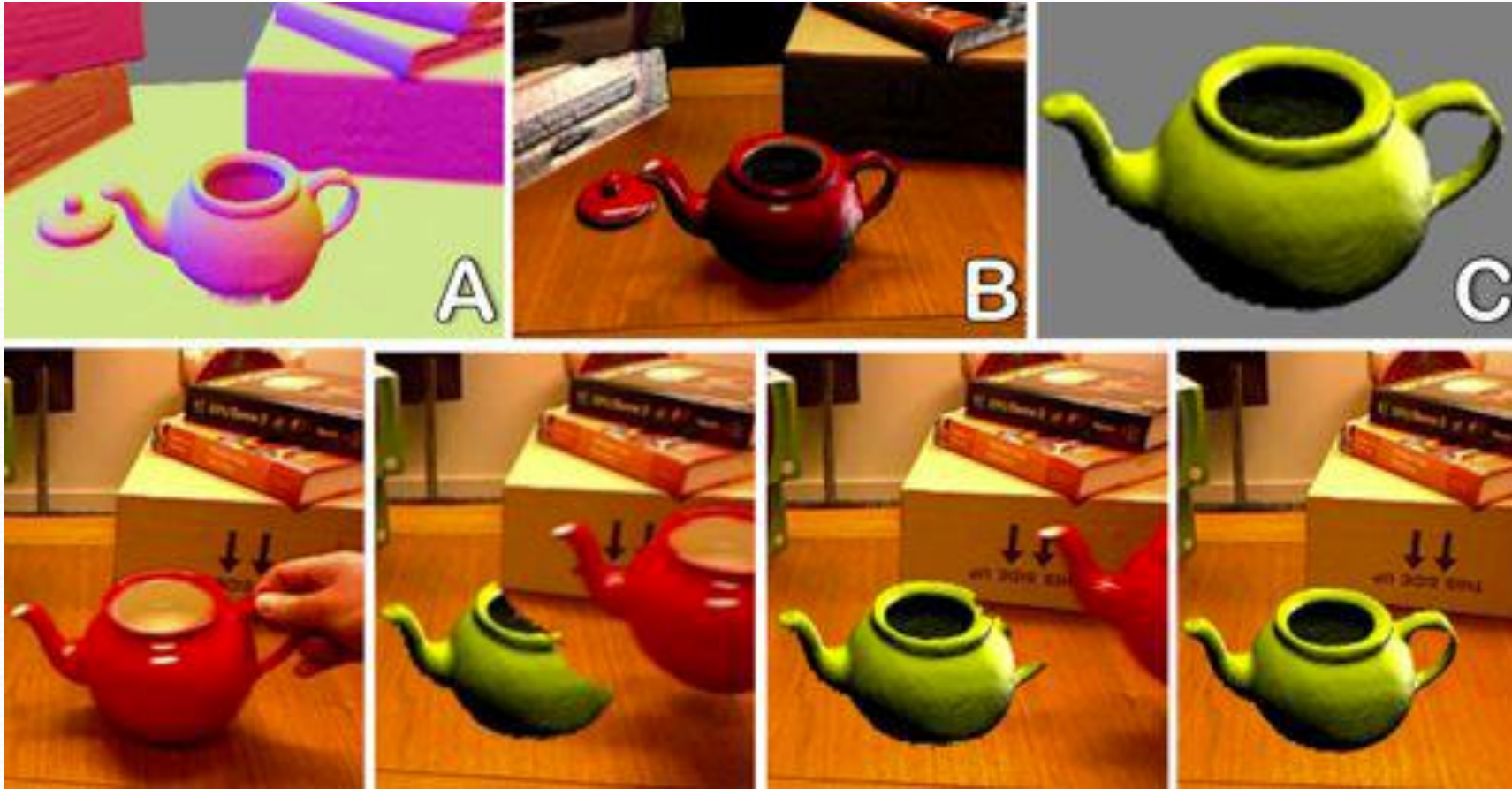
Dr. Cheung, committee member

- Presented By: Sen Li

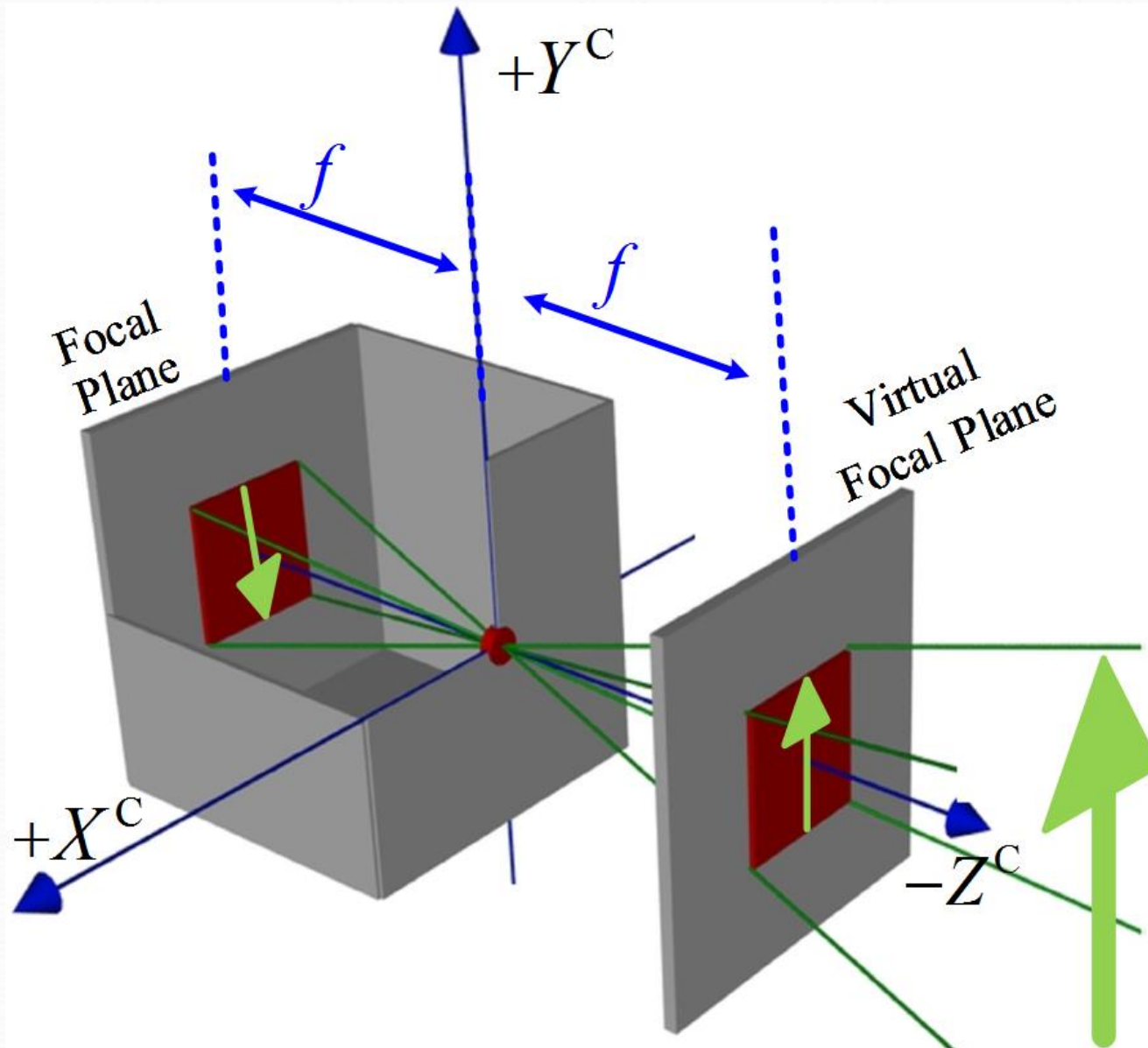
Outline

- **Introduction**
- Academic Background:
 - Pinhole Camera Model (Camera calibration / 3D reconstruction)
 - Kai's per-pixel 3D reconstruction on GPU
 - Distortions
- Per-Pixel Distortion Correction & Natural GPU Reconstruction:
 - Calibration system
 - Calibration procedures
 - Undistorted 3D Reconstruction
- Conclusion & Future Work

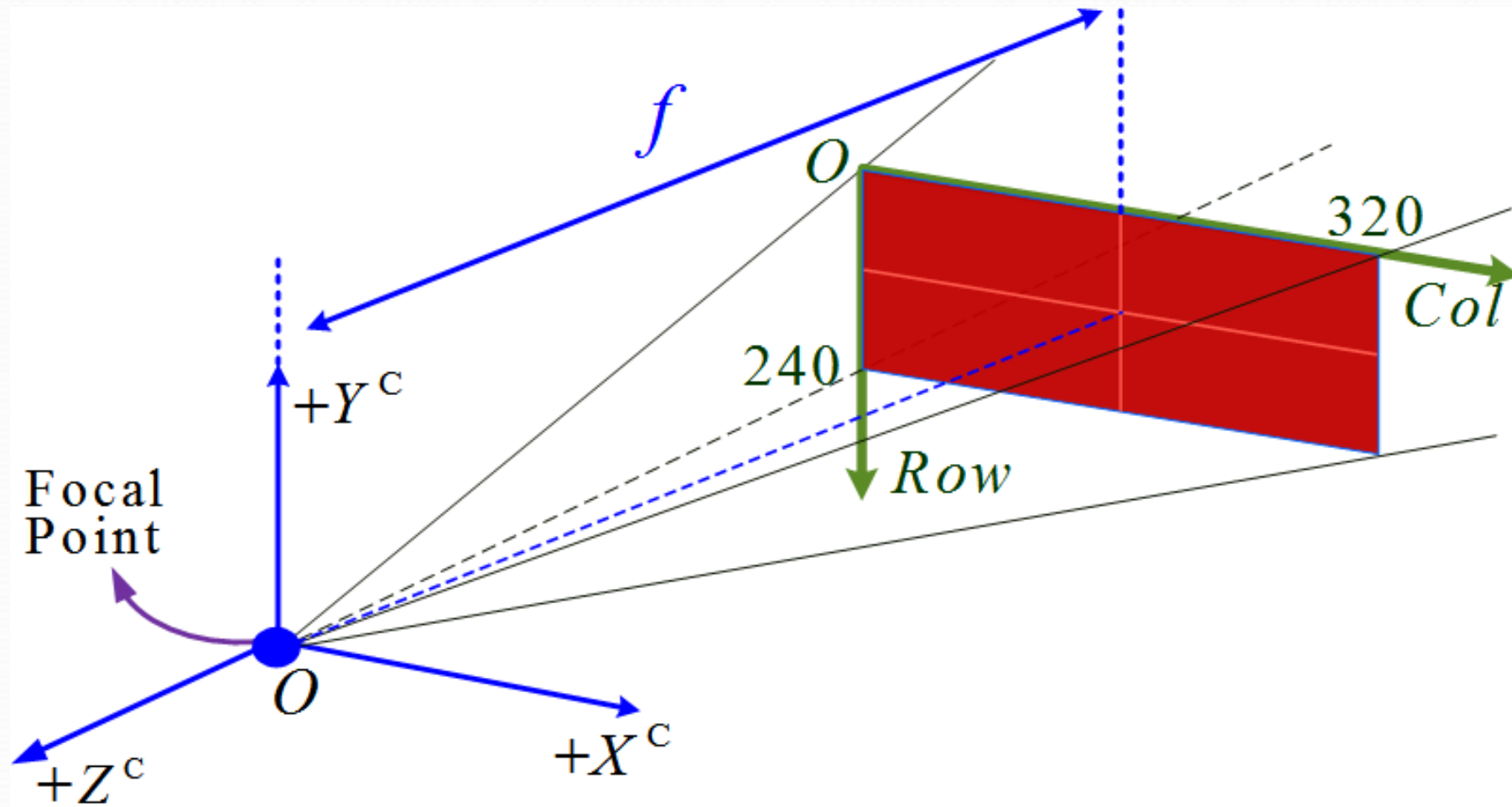
Object Segmentation in KinectFusion



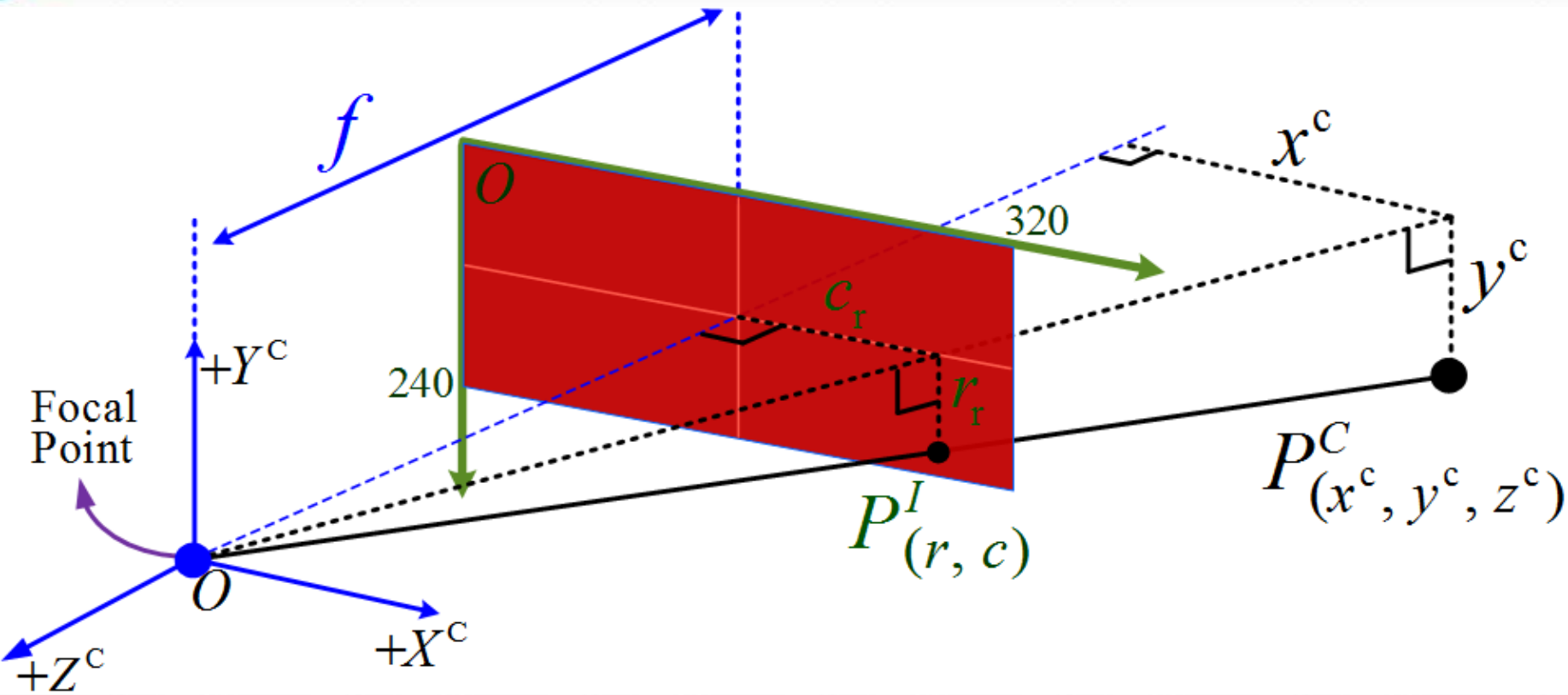
Pinhole-Camera Model



Pinhole-Camera Model



Pinhole-Camera Model



$$\begin{bmatrix} \alpha_c f & 0 & \alpha_c c_h \\ 0 & \alpha_r f & \alpha_r r_h \\ 0 & 0 & 1 \end{bmatrix}$$

$$K = \begin{bmatrix} f_c & s & t_c \\ 0 & f_r & t_r \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} c \\ r \end{bmatrix} = f \begin{bmatrix} x^c / z^c \\ y^c / z^c \end{bmatrix} + \begin{bmatrix} c_h \\ r_h \end{bmatrix}$$

$$z^c \begin{bmatrix} c \\ r \\ 1 \end{bmatrix} = \begin{bmatrix} f x^c \\ f y^c \\ z^c \end{bmatrix} + \begin{bmatrix} z^c c_h \\ z^c r_h \\ 0 \end{bmatrix} = \begin{bmatrix} f & 0 & c_h \\ 0 & f & r_h \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^c \\ y^c \\ z^c \end{bmatrix}$$

- Traditional Reconstruction (Calibrate \mathbf{M} and then Inverse)

$$M = K \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}$$

$$Z^C \begin{bmatrix} R \\ C \\ 1 \end{bmatrix} = \mathbf{M}_{3 \times 4} \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix} = [\mathbf{M}^T \mathbf{M}]^{-1} \mathbf{M}^T Z^C \begin{bmatrix} R \\ C \\ 1 \end{bmatrix}$$

- Kai's : Natural Shader Reconstruction on GPU

Pinhole Camera Model

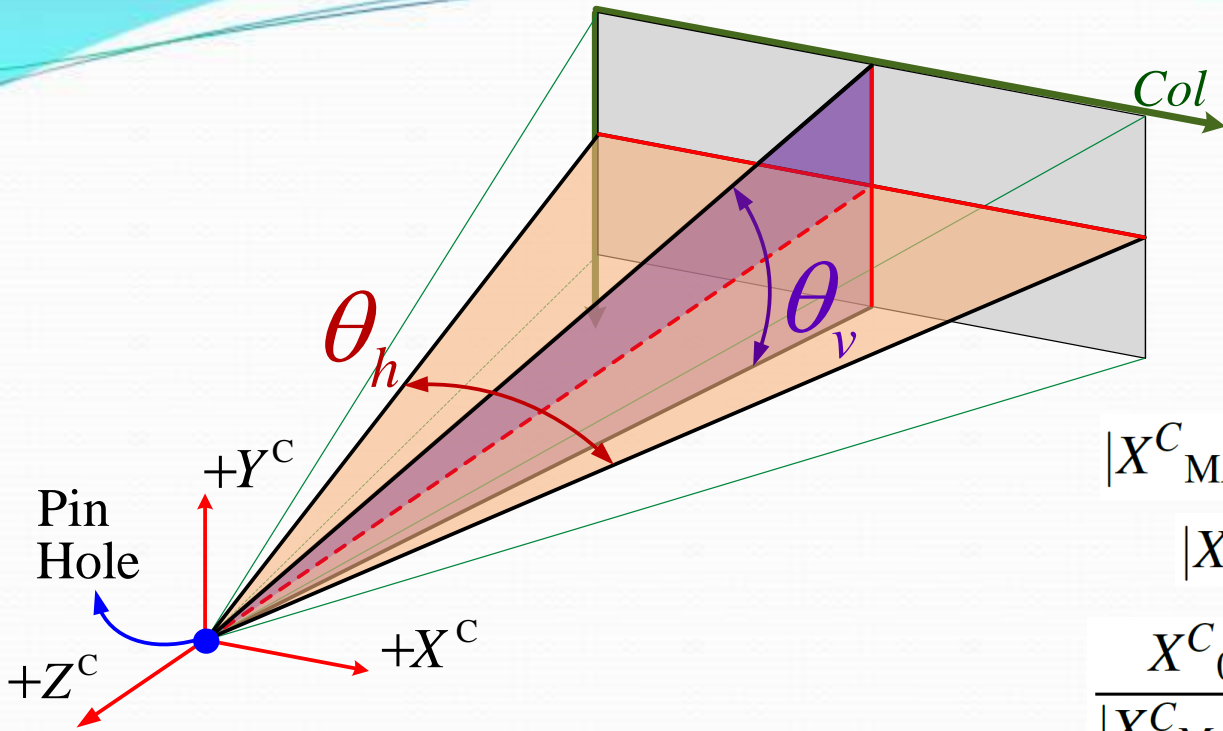
$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \rightarrow \begin{aligned} X^W[m, n] &= a[m, n]Z^W[m, n] + b[m, n] \\ Y^W[m, n] &= c[m, n]Z^W[m, n] + d[m, n] \end{aligned}$$

- KinectV2: Raw 3D Reconstruction on GPU based on θ_h / θ_v (Intrinsic \mathbf{K})

$$Z^C[m, n] = -\beta D[m, n] \quad \beta = 65535.0$$

$$\begin{aligned} \theta_h \\ \theta_v \end{aligned} \rightarrow \begin{aligned} X^C[m, n] &= a[m, n] \cdot |Z^C[m, n]| \\ Y^C[m, n] &= b[m, n] \cdot |Z^C[m, n]| \end{aligned}$$

- Raw 3D Reconstruction on GPU based on θ_h / θ_v (Intrinsic K)



$$Z^C[m, n] = -\beta D[m, n]$$

65535.0

$$X^C[m, n] = a[m, n] \cdot |Z^C[m, n]|$$

$$Y^C[m, n] = b[m, n] \cdot |Z^C[m, n]|$$

$$|X^C_{MAX}| = |Z^C_0| \cdot \tan(\theta_h/2)$$

$$|X^C_0| = |Z^C_0| \cdot \tan(\alpha_0)$$

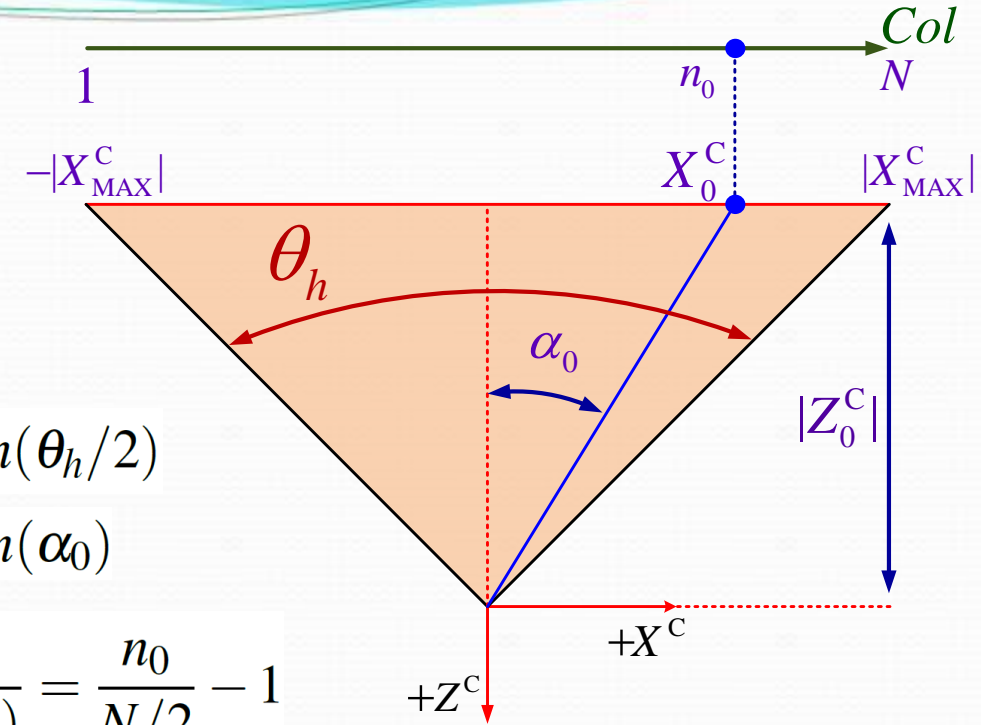
$$\frac{X^C_0}{|X^C_{MAX}|} = \frac{\tan(\alpha_0)}{\tan(\theta_h/2)} = \frac{n_0}{N/2} - 1$$

$$X^C[m, n] = \tan(\theta_h/2) \cdot \left(\frac{n}{N/2} - 1\right) \cdot |Z^C[m, n]|$$

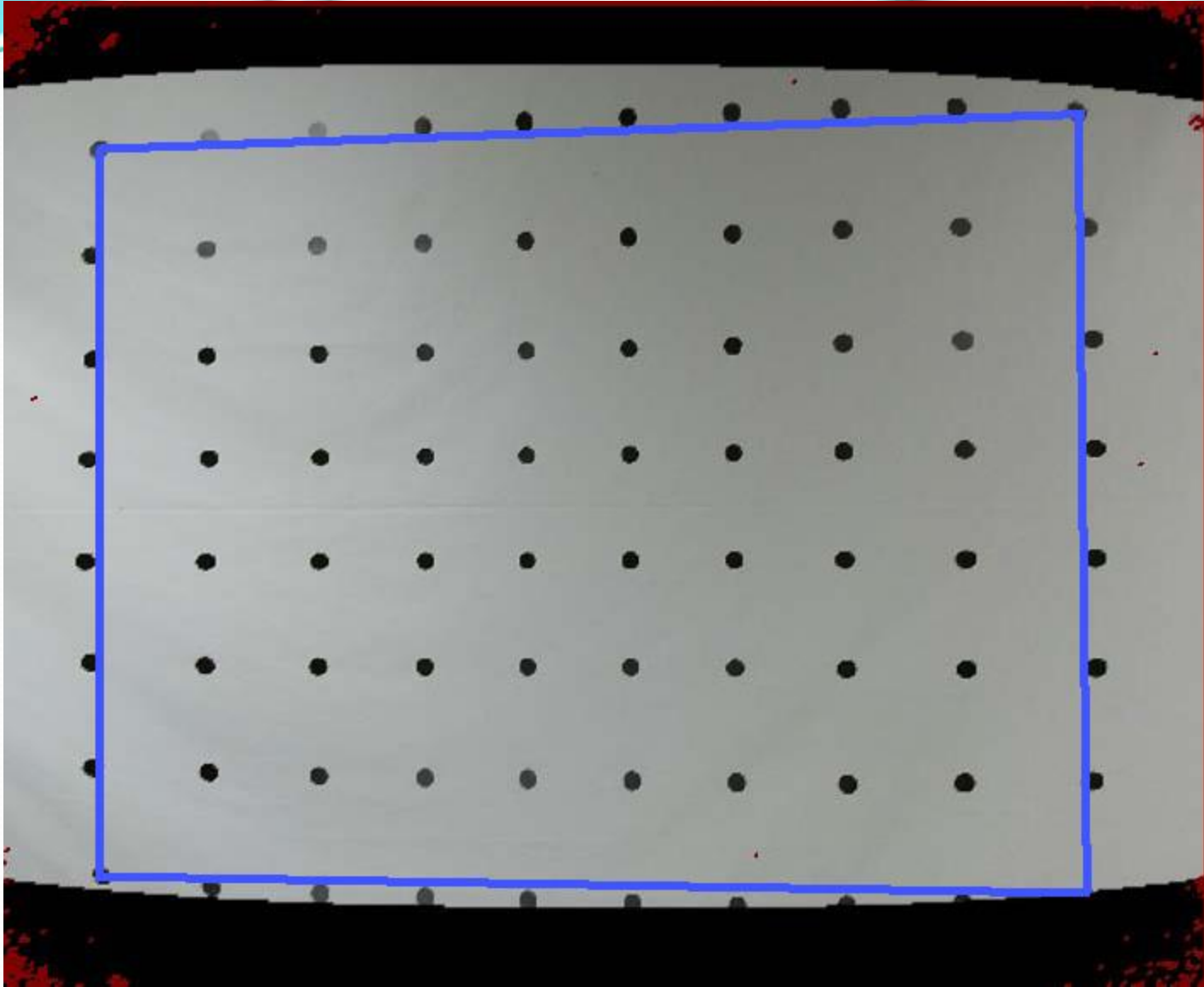
$$Y^C[m, n] = \tan(\theta_v/2) \cdot \left(\frac{m}{M/2} - 1\right) \cdot |Z^C[m, n]|$$

$$a[m, n] = \tan(\theta_h/2) \cdot \left(\frac{n}{N/2} - 1\right)$$

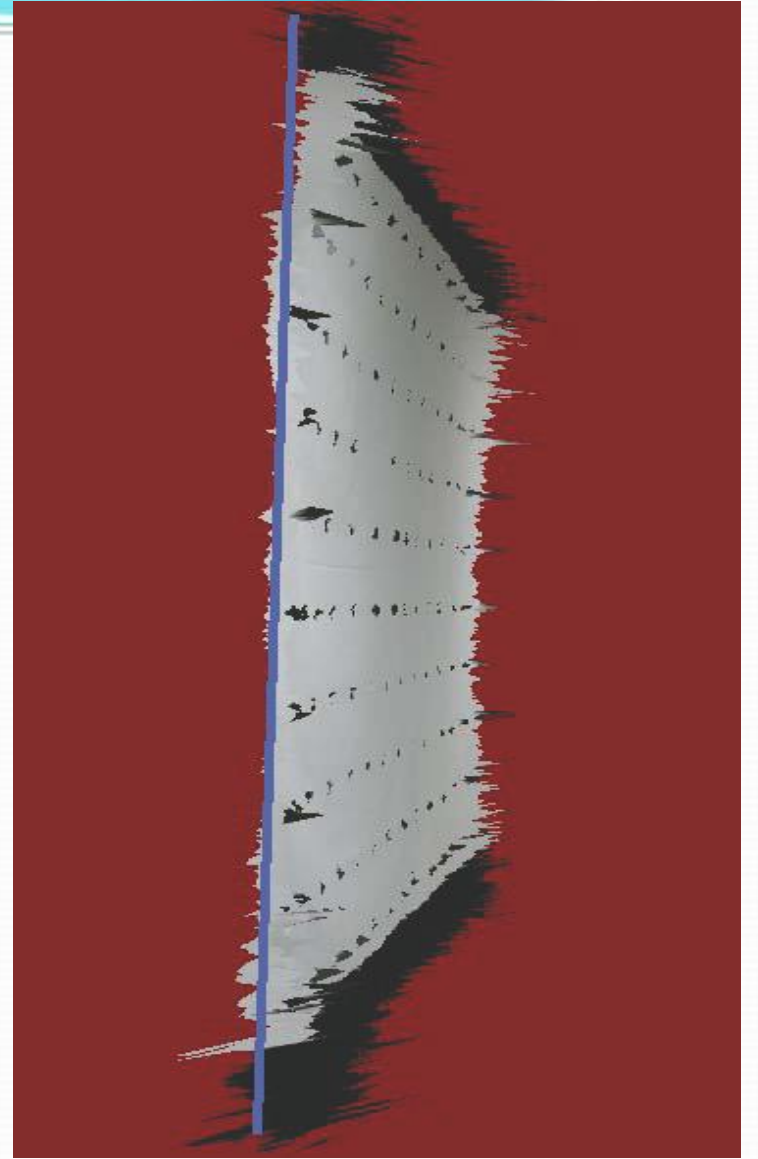
$$b[m, n] = \tan(\theta_v/2) \cdot \left(\frac{m}{M/2} - 1\right)$$



Raw KinectV2 3D Reconstruction

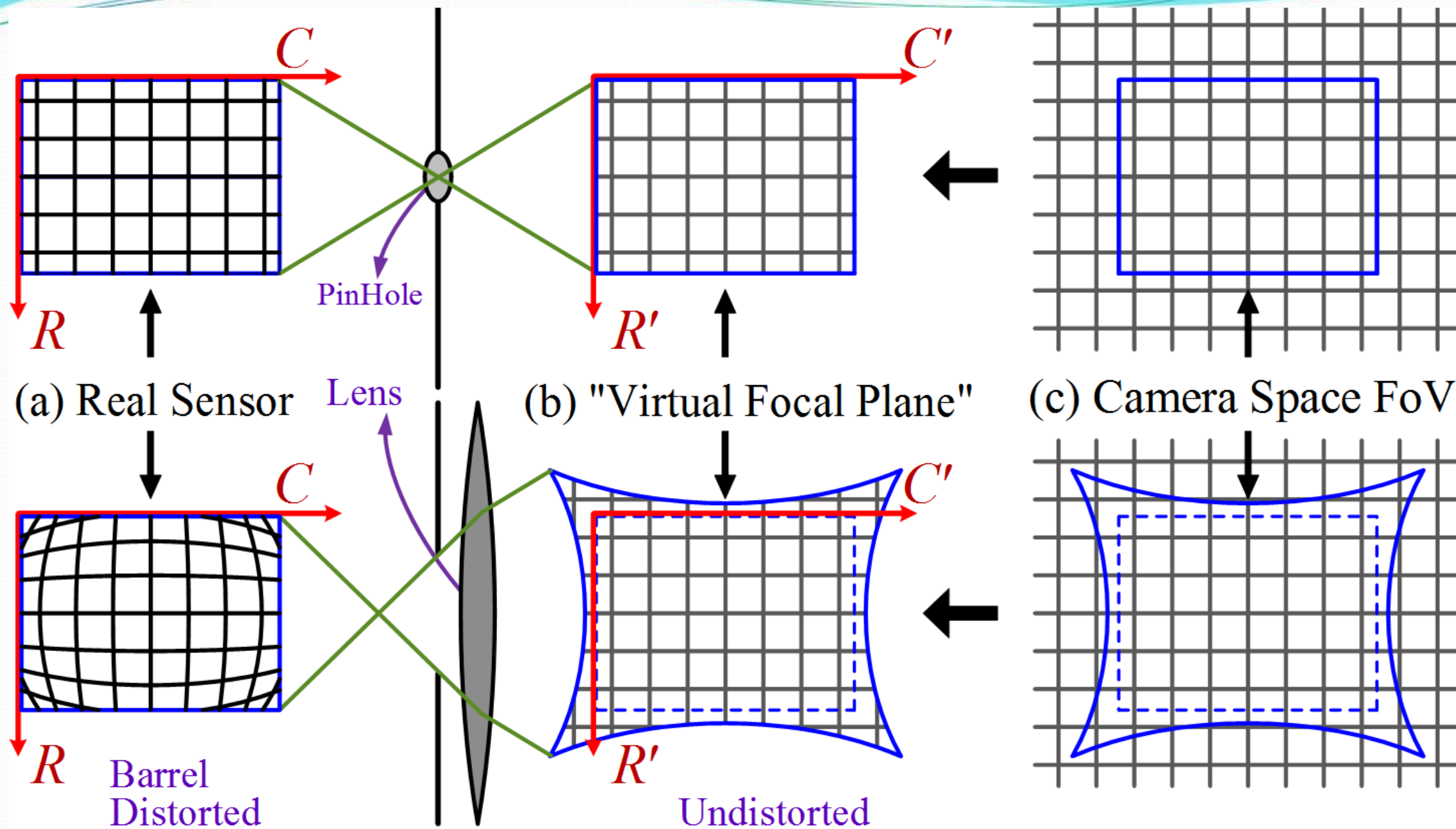


lens Distortion



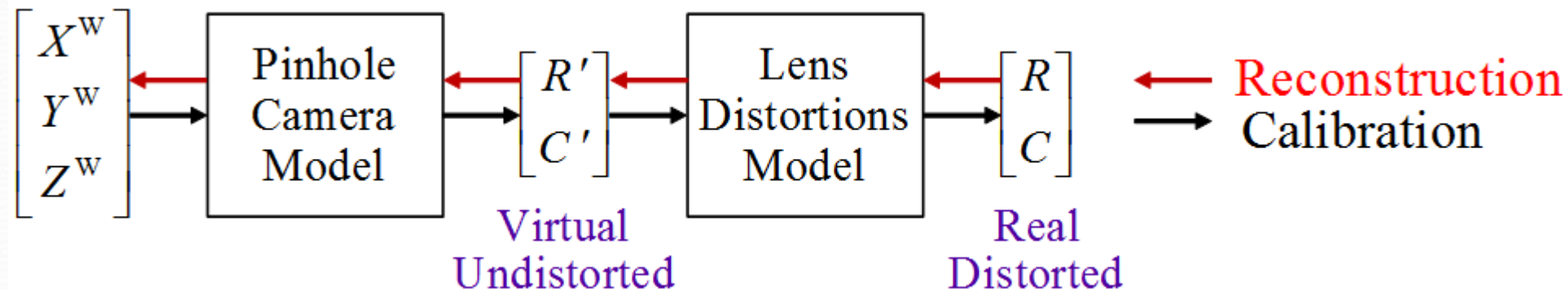
Depth Distortion

- From Camera Space to Image Space with Lens Distortions



Academic Background

- **Pinhole-Camera Model** (Calibration and Reconstruction)



Intrinsic Matrix

$$K = \begin{bmatrix} f_c & s & t_c \\ 0 & f_r & t_r \\ 0 & 0 & 1 \end{bmatrix}$$

Extrinsic Matrix

$$\begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \end{bmatrix}$$

Pinhole Camera Matrix :

$$M = K \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}$$

$$C' = C(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + [p_1(r^2 + 2C^2) + 2p_2 CR]$$

$$R' = R(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + [p_2(r^2 + 2R^2) + 2p_1 CR]$$

Distortion Parameters: $k_1/k_2/k_3/p_1/p_2$

$$Z^C \begin{bmatrix} R' \\ C' \\ 1 \end{bmatrix} = M_{3 \times 4} \begin{bmatrix} X^w \\ Y^w \\ Z^w \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} X^w \\ Y^w \\ Z^w \\ 1 \end{bmatrix} = [M^T M]^{-1} M^T Z^C \begin{bmatrix} R' \\ C' \\ 1 \end{bmatrix}$$

- Traditional Undistorted Reconstruction (High Order and Inverse)

Distortion Parameters: $k_1/k_2/k_3/p_1/p_2$

$$\begin{aligned} C' &= C(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + [p_1(r^2 + 2C^2) + 2p_2 CR] \\ R' &= R(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + [p_2(r^2 + 2R^2) + 2p_1 CR] \end{aligned}$$

$$k \begin{bmatrix} R' \\ C' \\ 1 \end{bmatrix} = \mathbf{M}_{3 \times 4} \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix} = [\mathbf{M}^T \mathbf{M}]^{-1} \mathbf{M}^T \begin{bmatrix} kR' \\ kC' \\ k \end{bmatrix}$$

- Kai's : "Natural" Shader Reconstruction ($k_1/k_2/k_3/p_1/p_2$ High Order)

$$\begin{aligned} m' &= m * (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + [p_1(r^2 + 2m^2) + 2p_2 * m * n] \\ n' &= n * (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + [p_1(r^2 + 2n^2) + 2p_2 * m * n] \end{aligned}$$

$$\begin{aligned} X^W[m, n] &= a[m', n'] Z^W[m, n] + b[m', n'] \\ Y^W[m, n] &= c[m', n'] Z^W[m, n] + d[m', n'] \end{aligned}$$

- Want: Undistorted Natural 3D Reconstruction on GPU

$$Z^C[m, n] = -\beta D[m, n]$$



65535.0

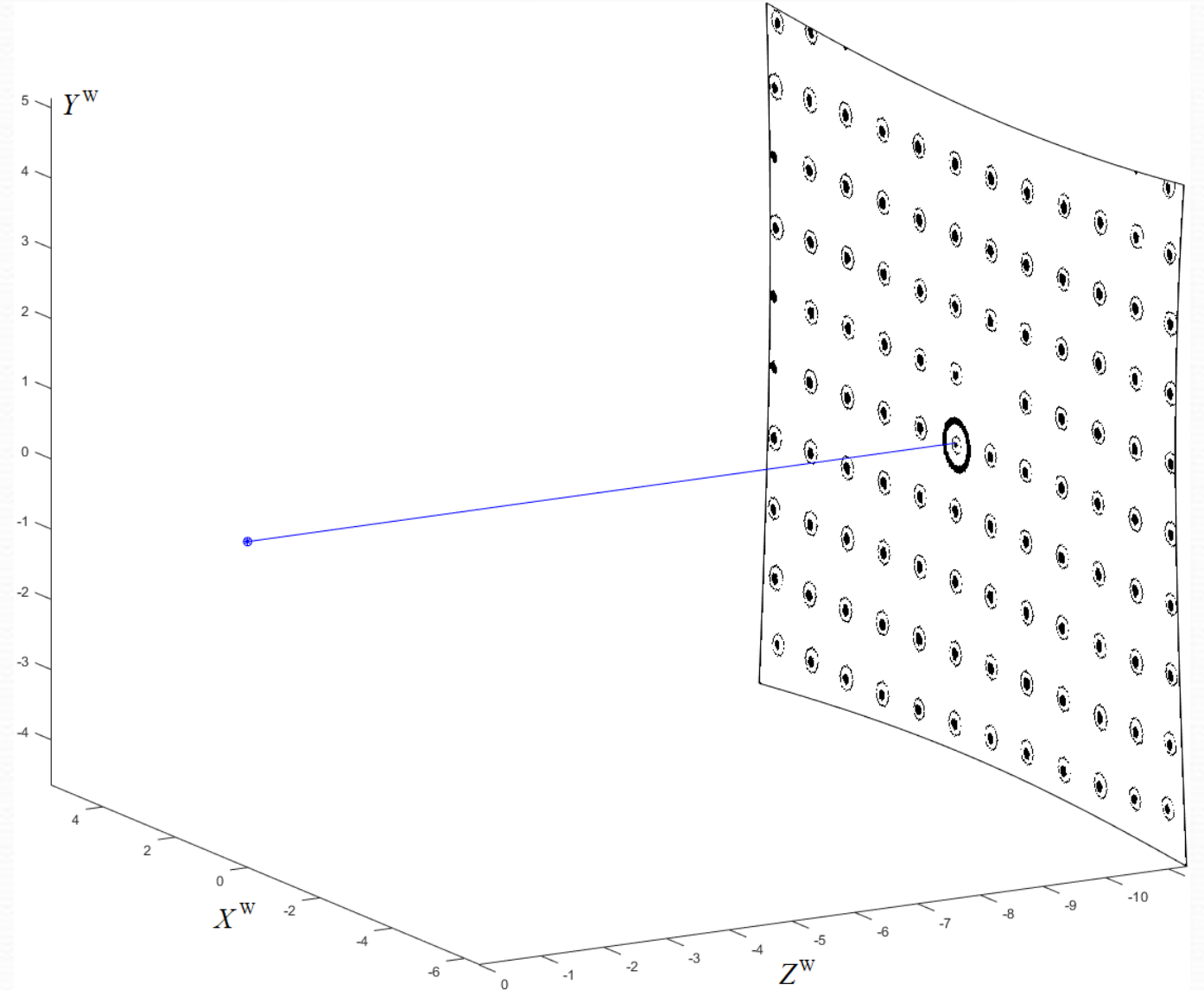
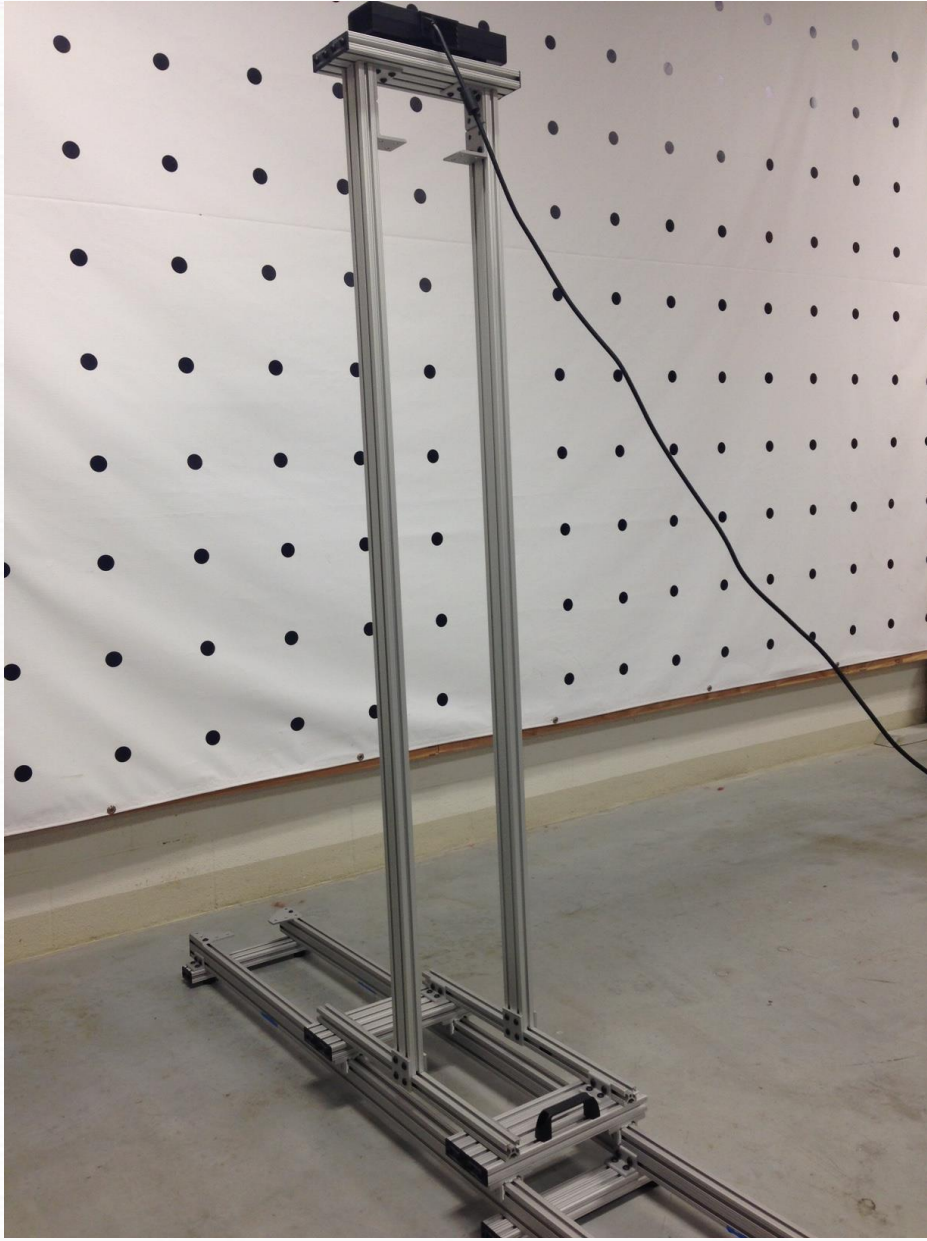
$$Z^W[m, n] = e[m, n] D[m, n] + f[m, n]$$

$$X^W[m, n] = a[m, n] Z^W[m, n] + b[m, n]$$

$$Y^W[m, n] = c[m, n] Z^W[m, n] + d[m, n]$$

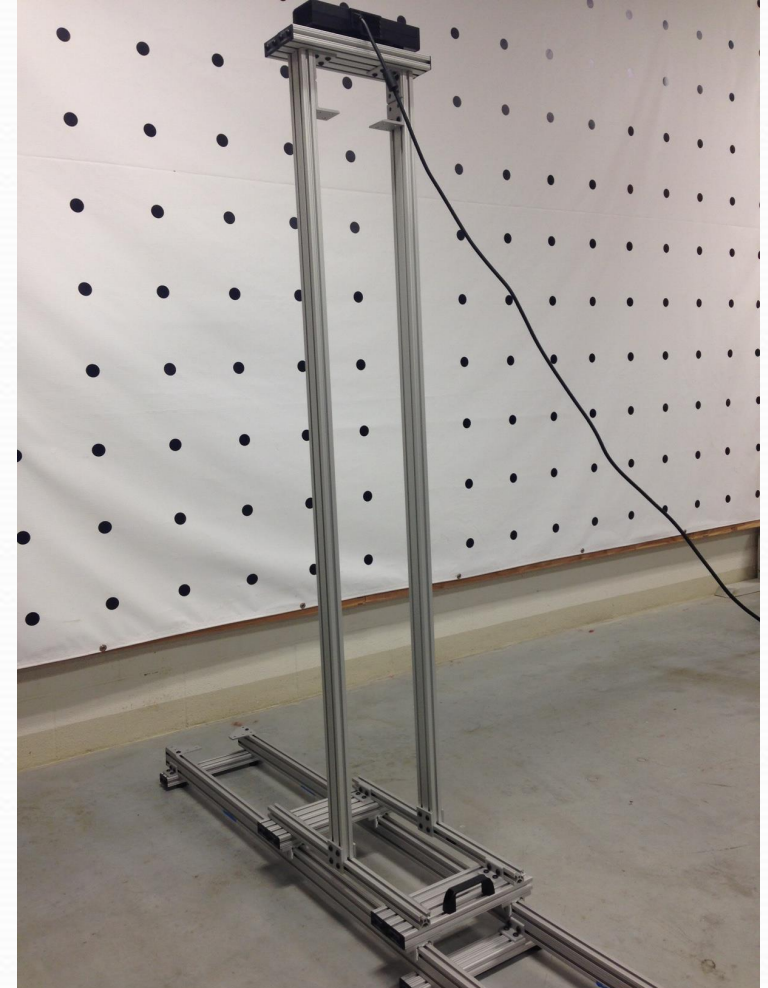
- Inspiration: find a way to get per-pixel Z^W

Rail Calibration System



Data Collection

- Mount: camera and laser distance measurer
- Want:
 - NIR: $X^w Y^w Z^w ID$
 - RGB: $X^w Y^w Z^w RGBD$
- Bound:
 - Z^w : Laser Distance Measurer
 - $X^w Y^w$: Uniform Round Dot Pattern
- Found (algorithms):
 - Calibration Points (*Row*, *Col*)s Extraction;
 - Corresponding World Space Address Assignment;
 - Non-Linear Dense Transformation



Calibration Points (*Row, Col*) Extraction;

- Gray-Scaling (luminosity method)

$$I_0[m, n] = 0.21R[m, n] + 0.72G[m, n] + 0.07B[m, n]$$

- Histogram Equalization

$$I_1[m, n] = \frac{I_0[m, n] - I_{\text{MIN}}}{I_{\text{MAX}} - I_{\text{MIN}}}$$

- Adaptive Thresholding

$$I_2[m, n] = \begin{cases} 1, & I_1[m, n] - I_b[m, n] - C_0 > 0 \\ 0, & \text{else} \end{cases}$$

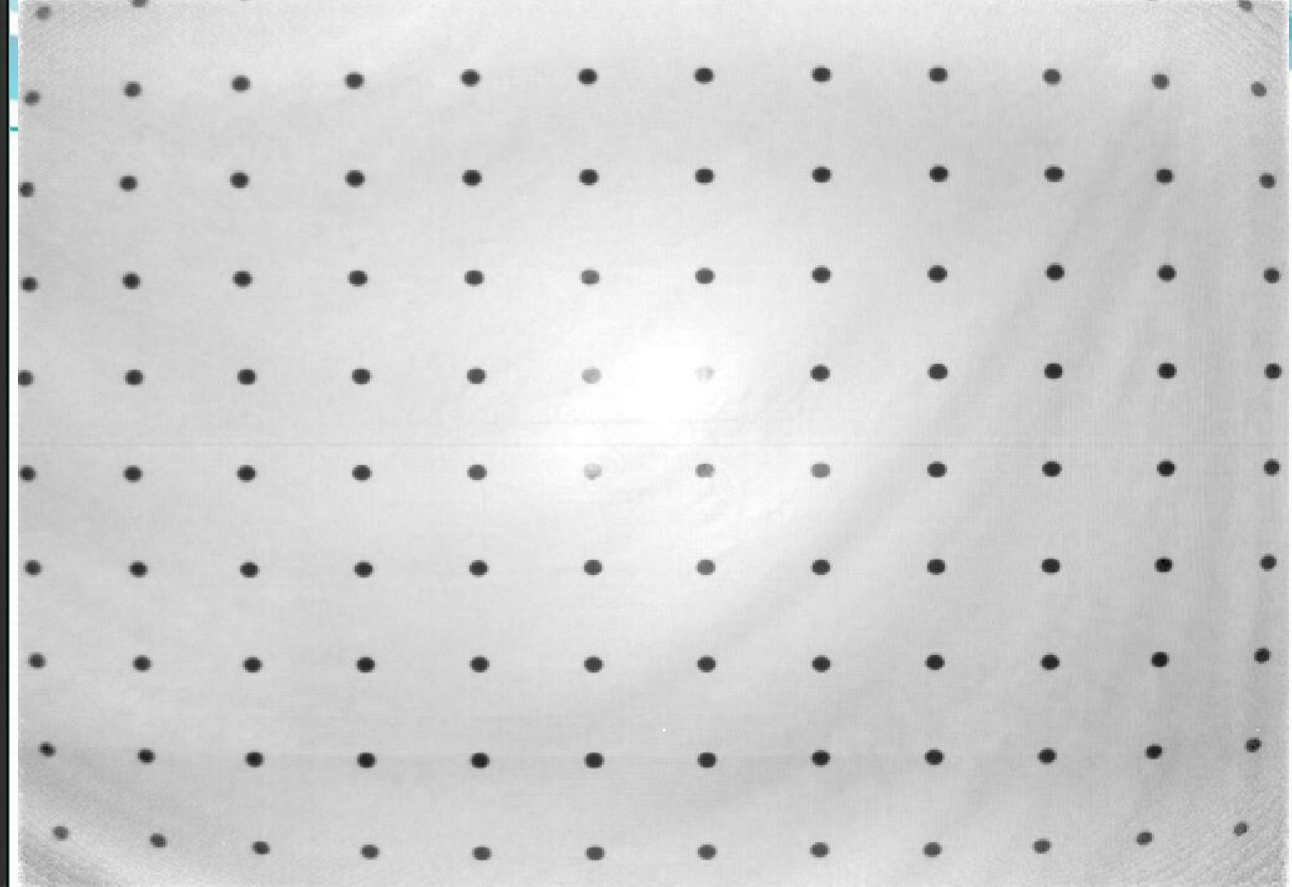
- Round Dot Tracking

<i>E</i>	<i>A</i>	<i>F</i>
<i>B</i>	<i>O</i>	<i>C</i>
<i>G</i>	<i>D</i>	<i>H</i>

$$I_3[m, n] = I_2[m, n] * \frac{(128I_A + 64I_B + 32I_C + 16I_D + 8I_E + 4I_F + 2I_G + I_H)}{255}$$



Raw NIR Image



After Histogram Equalization

$$I_0[m, n] = \frac{g[m, n]}{G} * (1.0 - 0.0) + 0.0 = \frac{g[m, n]}{G}$$

$$CDF(g) = \frac{\sum_{l=1}^g PMF[l]}{M \times N}$$

$$CDF(g_{\min}) = 0.01$$

$$CDF(g_{\max}) = 0.99$$

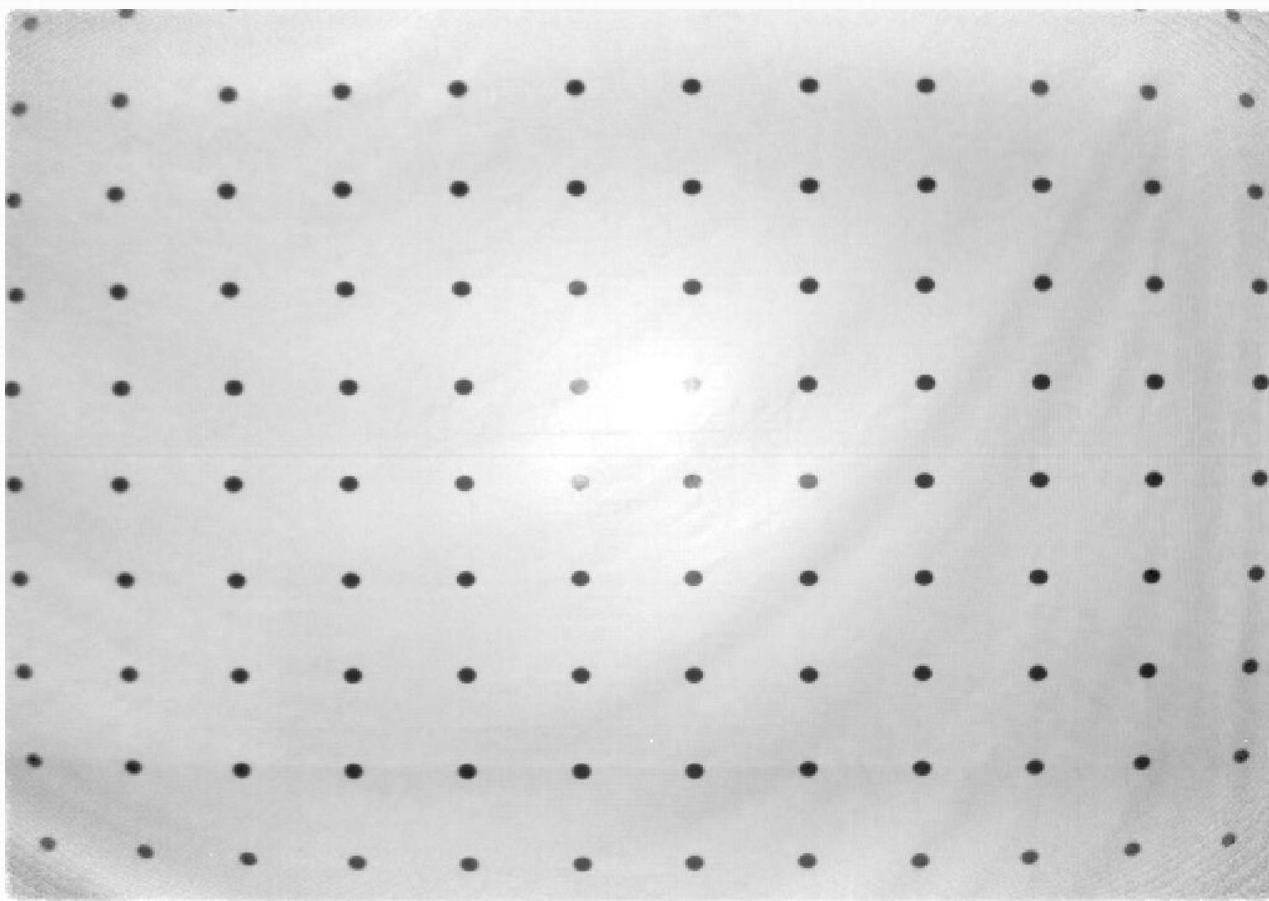
$$I_{\min} = g_{\min} / G$$

$$I_{\max} = g_{\max} / G$$

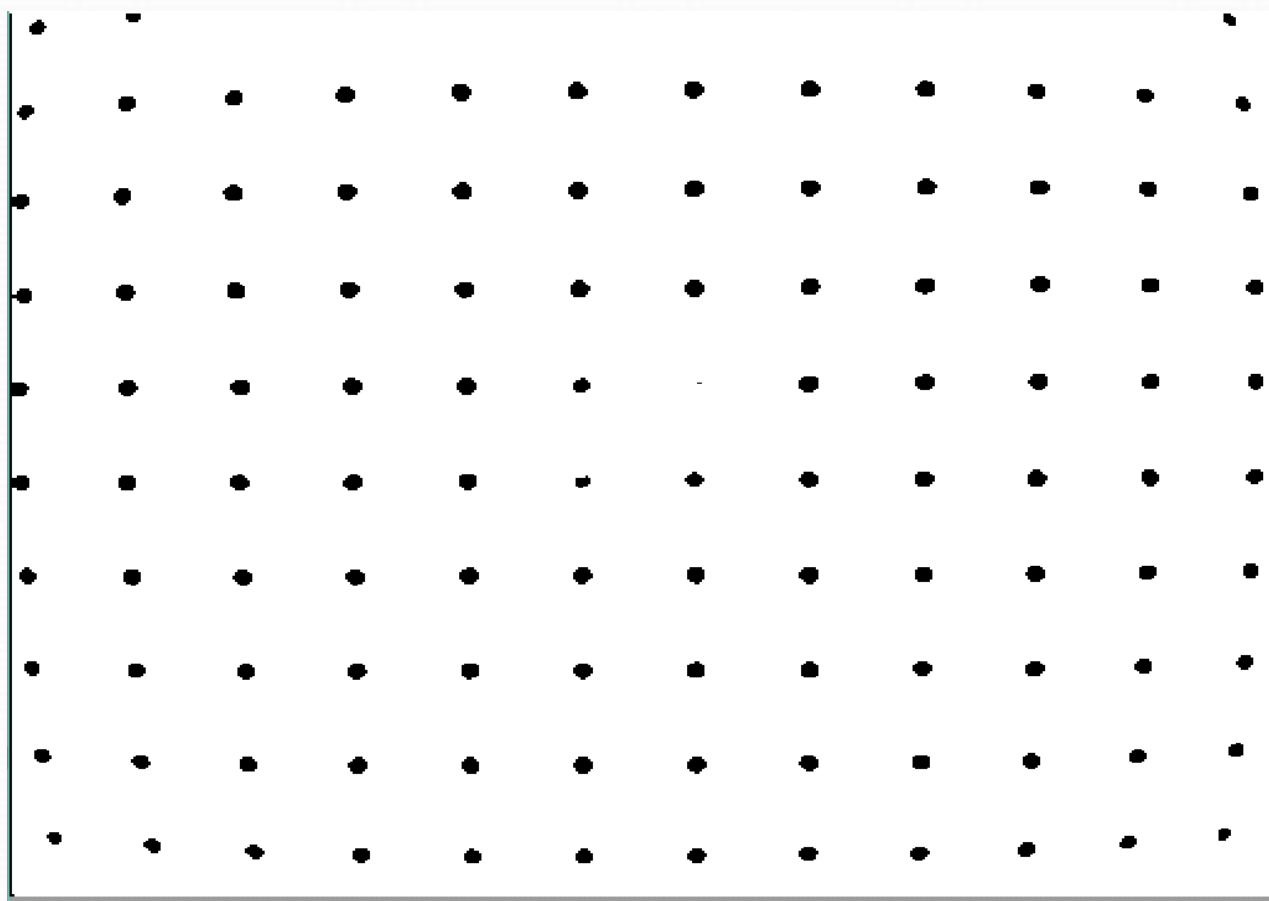
$$I_1[m, n] = \frac{I_0[m, n] - I_{\min}}{I_{\max} - I_{\min}}$$

- Adaptive Thresholding

$$I_2[m,n] = \begin{cases} 1, & I_1[m,n] - I_b[m,n] - C_0 > 0 \\ 0, & \text{else} \end{cases}$$



Histogram Equalized



Binarized

- Round Dot Tracking

<i>E</i>	<i>A</i>	<i>F</i>
<i>B</i>	<i>O</i>	<i>C</i>
<i>G</i>	<i>D</i>	<i>H</i>

$$I_3[m, n] = I_2[m, n] * \frac{(128I_A + 64I_B + 32I_C + 16I_D + 8I_E + 4I_F + 2I_G + I_H)}{255}$$

if ($I_O \& 0x80 == 1$), *then*, marker *A* is valid (go Up)

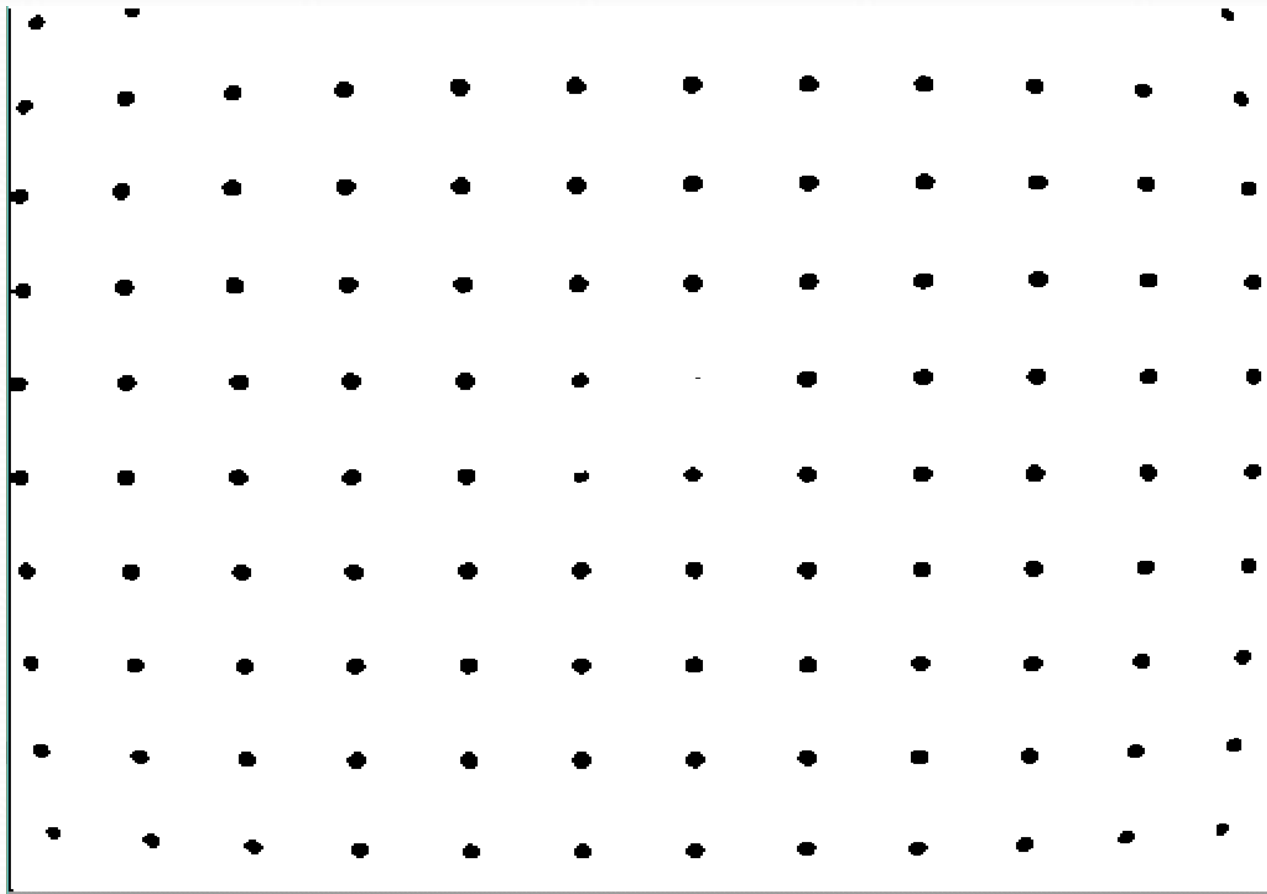
if ($I_O \& 0x40 == 1$), *then*, marker *B* is valid (go Left)

if ($I_O \& 0x20 == 1$), *then*, marker *C* is valid (go Right)

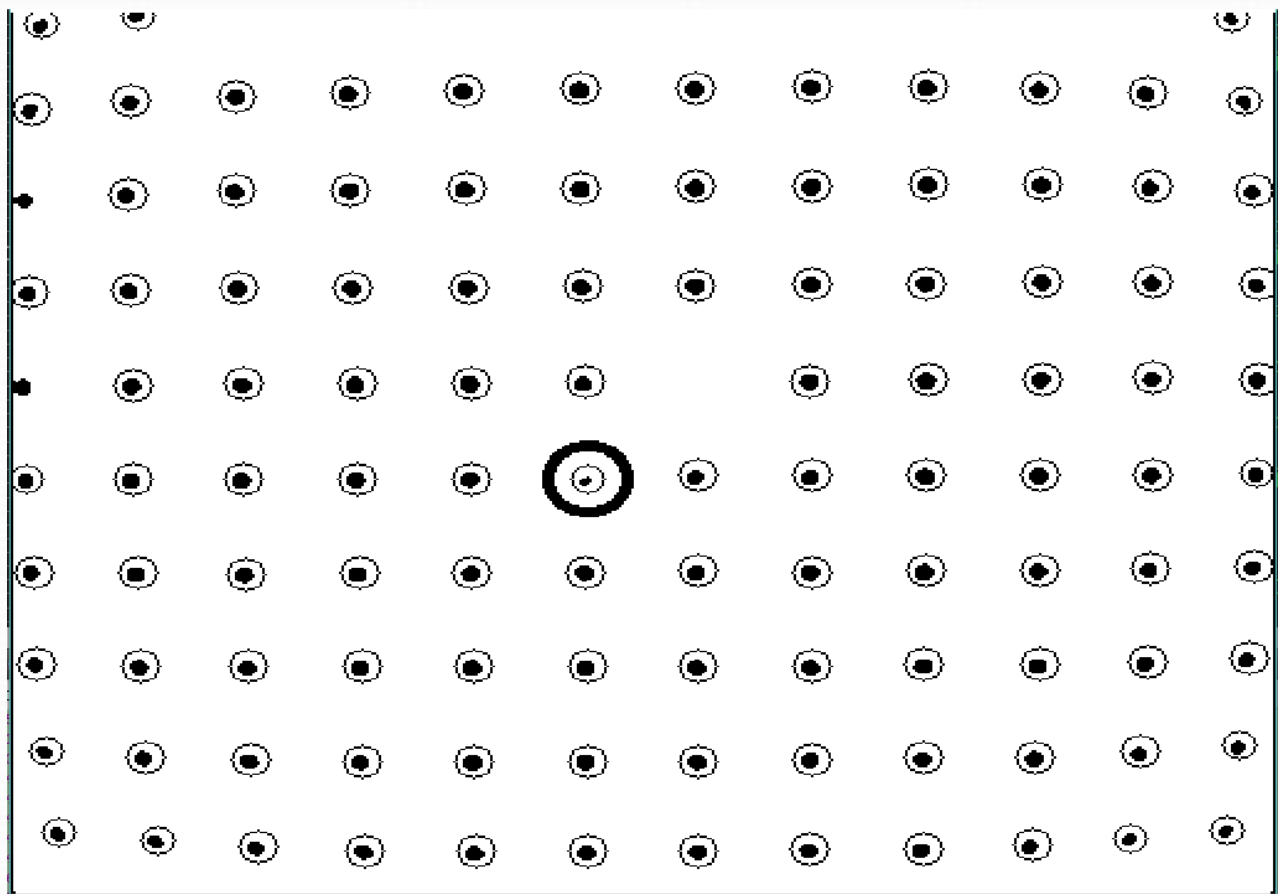
if ($I_O \& 0x10 == 1$), *then*, marker *D* is valid (go Down)

Anchor !=0 → Detect Valid → Flip Valid → Get Area + Bounding Box

Traverse and Extract on CPU



Binarized Image



Dots' Centers Extracted

World Space Address Assignment

- Travers every 4 Square-Shaped points and find best A_0 , which corresponds to the best-fit “Unit One” (228mm) in Image Space:

Get Linear Mapping matrix $A_0 \rightarrow$ Transform all (R, C) to (X^W, Y^W)

\rightarrow Count (N_v) valid points with integer $X^W/Y^W \rightarrow$ Find best A_0 who generates the largest N_v .

$$\begin{array}{cc} (0, 1) & (1, 1) \\ \bullet & \bullet \\ \bullet & \bullet \\ (0, 0) & (1, 0) \end{array} \quad \begin{bmatrix} zX^W \\ zY^W \\ z \end{bmatrix} = A_0 \begin{bmatrix} C \\ R \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} C \\ R \\ 1 \end{bmatrix}$$

- Translate Origin to the Center Dot

$$\begin{aligned} c_h &= (512-1) / 2 = 255.5 \\ r_h &= (424-1) / 2 = 211.5 \end{aligned} \quad \begin{bmatrix} zx_h \\ zy_h \\ z \end{bmatrix} = A_0 \cdot \begin{bmatrix} c_h \\ r_h \\ 1 \end{bmatrix} \quad \begin{aligned} c_h &= \text{round}(c_h) \\ r_h &= \text{round}(r_h) \end{aligned} \quad A_1 = T \cdot A_0 = \begin{bmatrix} 1 & 0 & -x_h \\ 0 & 1 & -y_h \\ 0 & 0 & 1 \end{bmatrix} \cdot A_0$$

World Space Address Assignment

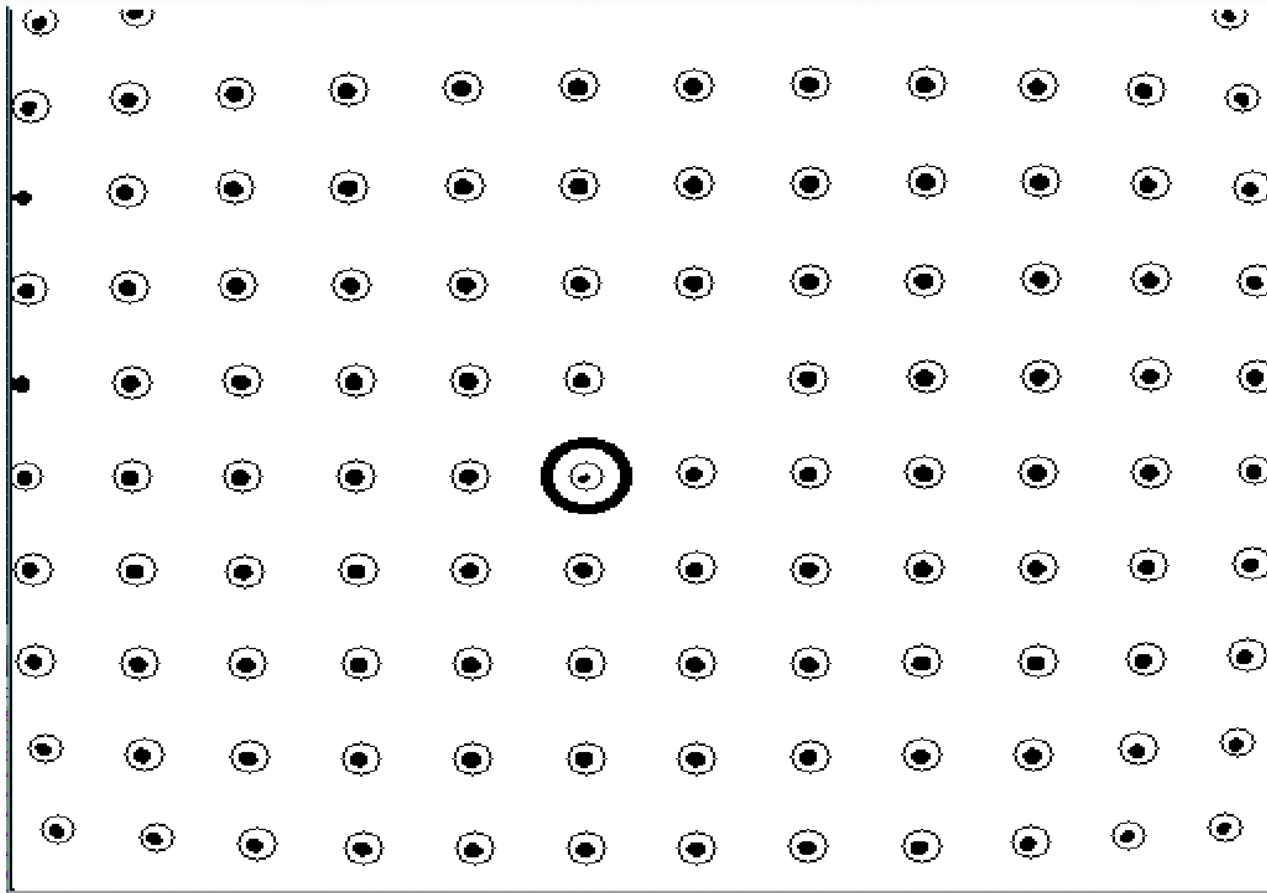
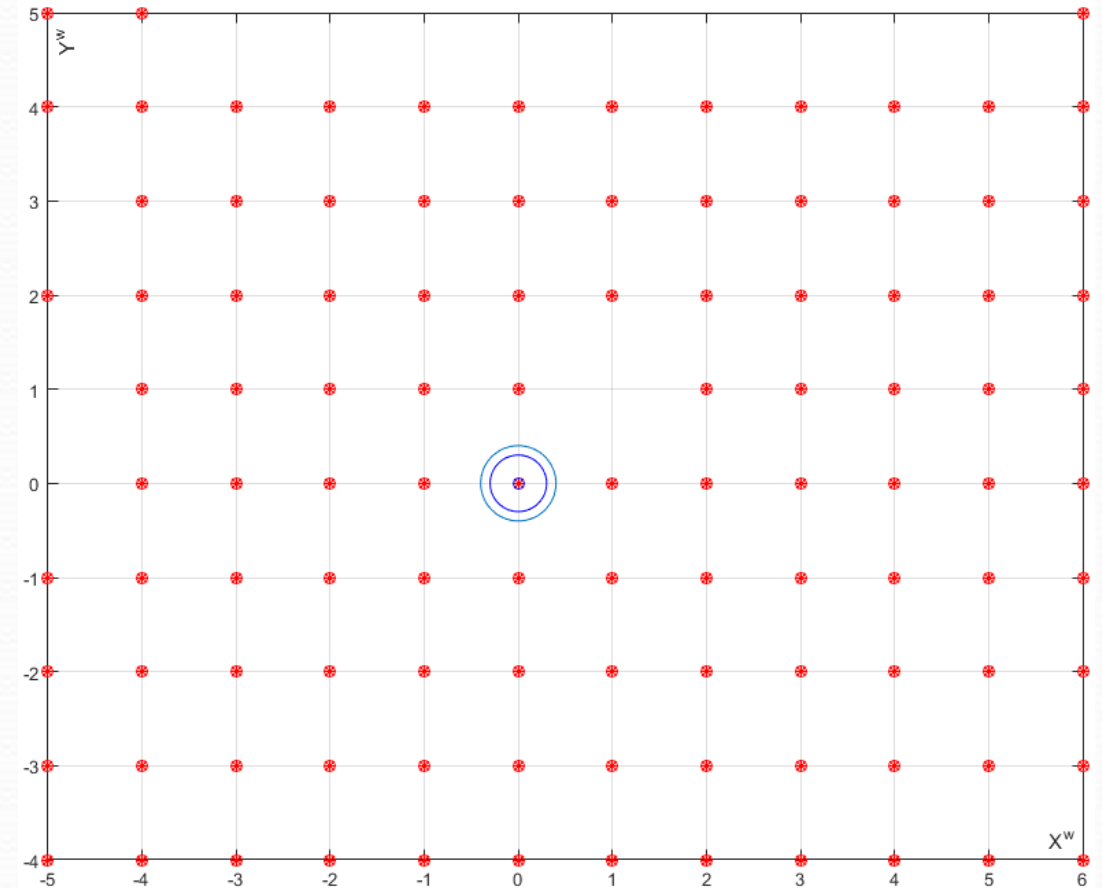


Image Space Calibration Points



Assigned World Space $X^w Y^w$

Two Dimensional Polynomial Mapping

- (First Order Perspective Transformation)

$$\begin{bmatrix} zX^W \\ zY^W \\ z \end{bmatrix} = A_0 \begin{bmatrix} C \\ R \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} C \\ R \\ 1 \end{bmatrix}$$

- Second Order

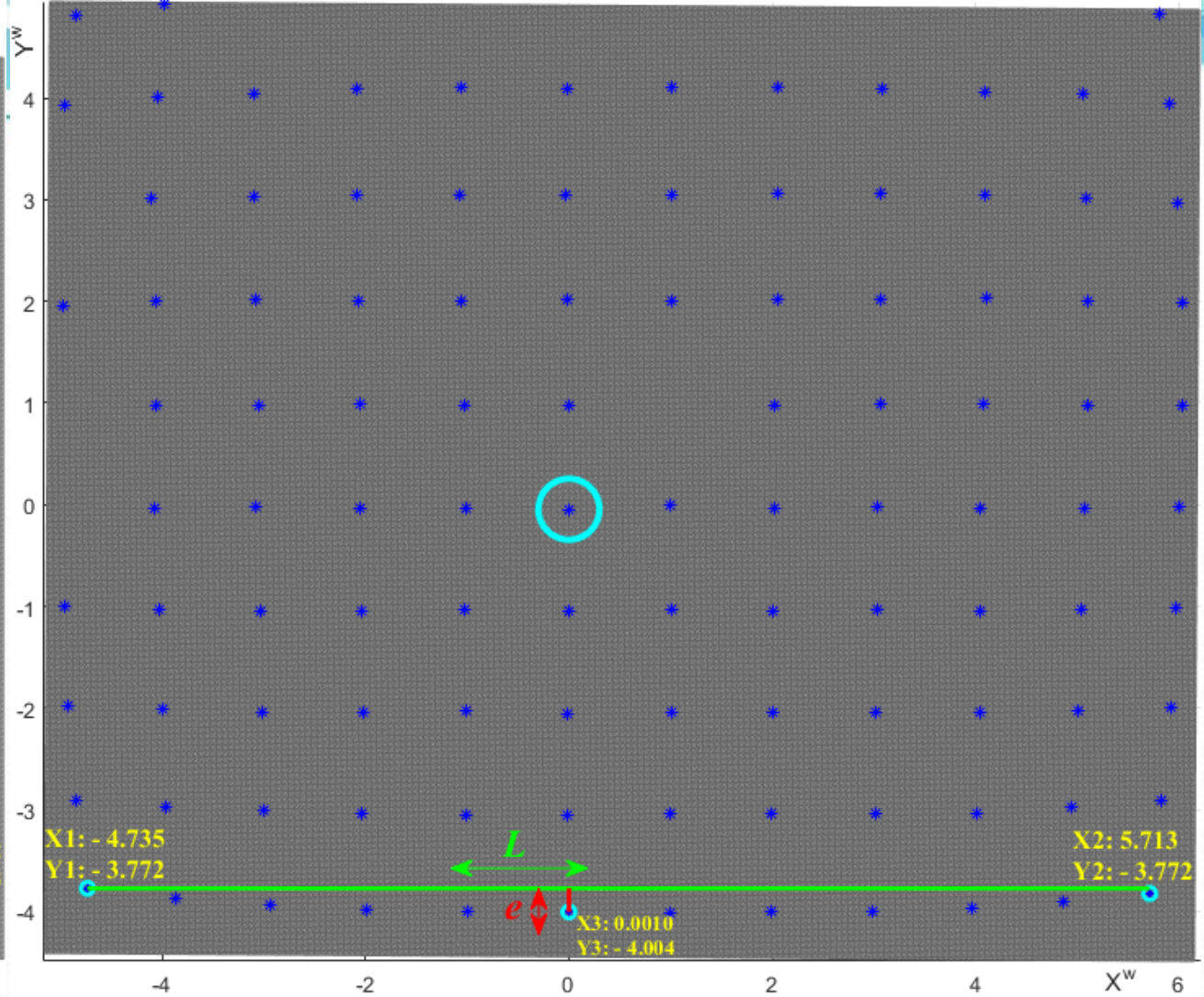
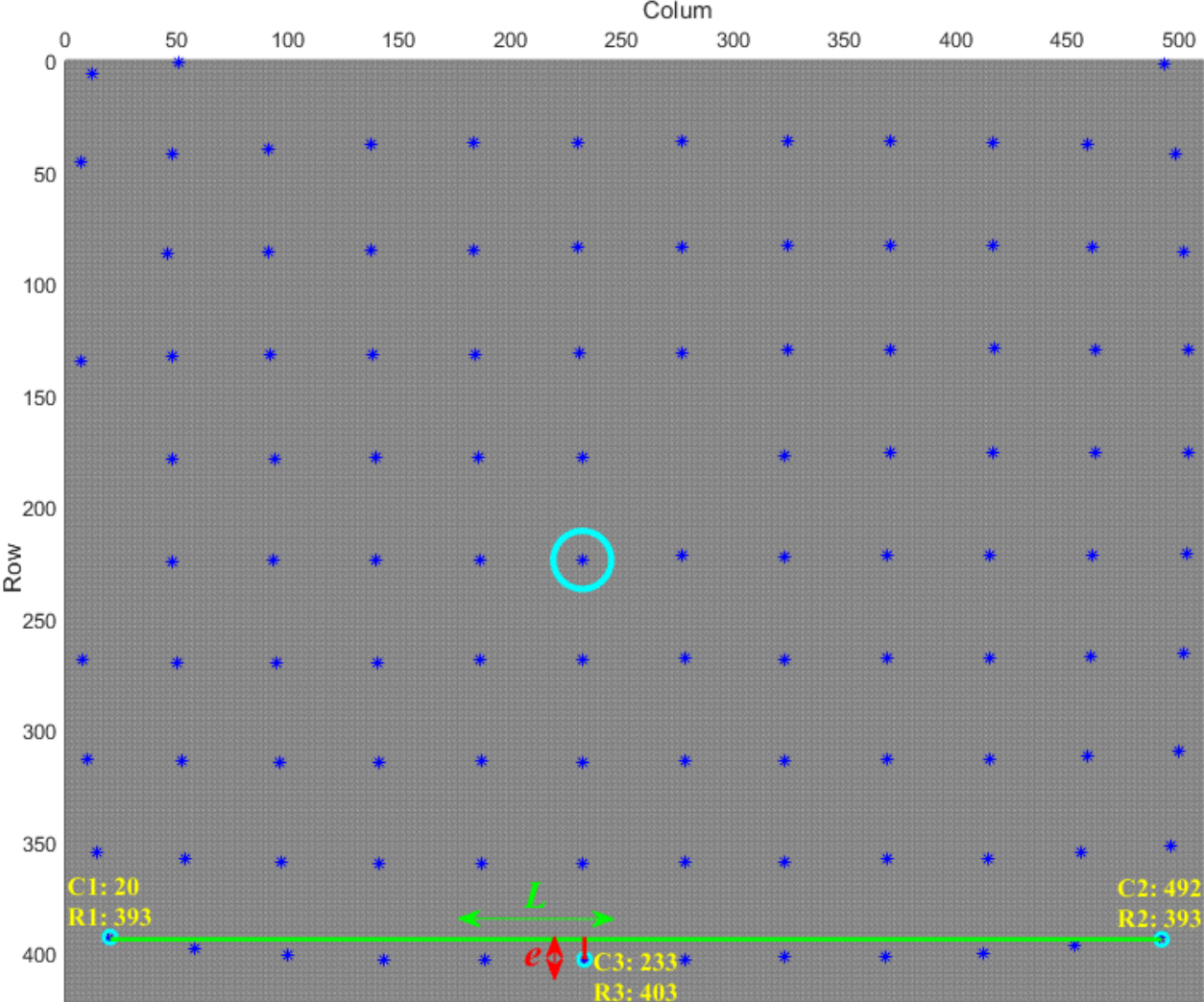
$$X^W = a_{11}C^2 + a_{12}CR + a_{13}R^2 + a_{14}C + a_{15}R + a_{16}$$

$$Y^W = a_{21}C^2 + a_{22}CR + a_{23}R^2 + a_{24}C + a_{25}R + a_{26}$$

- Fourth Order

$$\begin{aligned} X^W = & a_{11}C^4 + a_{12}C^3R + a_{13}C^2R^2 + a_{14}CR^3 + a_{15}R^4 + a_{16}C^3 + a_{17}C^2R \dots \\ & + a_{18}CR^2 + a_{19}R^3 + a_{110}C^2 + a_{111}CR + a_{112}R^2 + a_{113}C + a_{114}R + a_{115} \end{aligned}$$

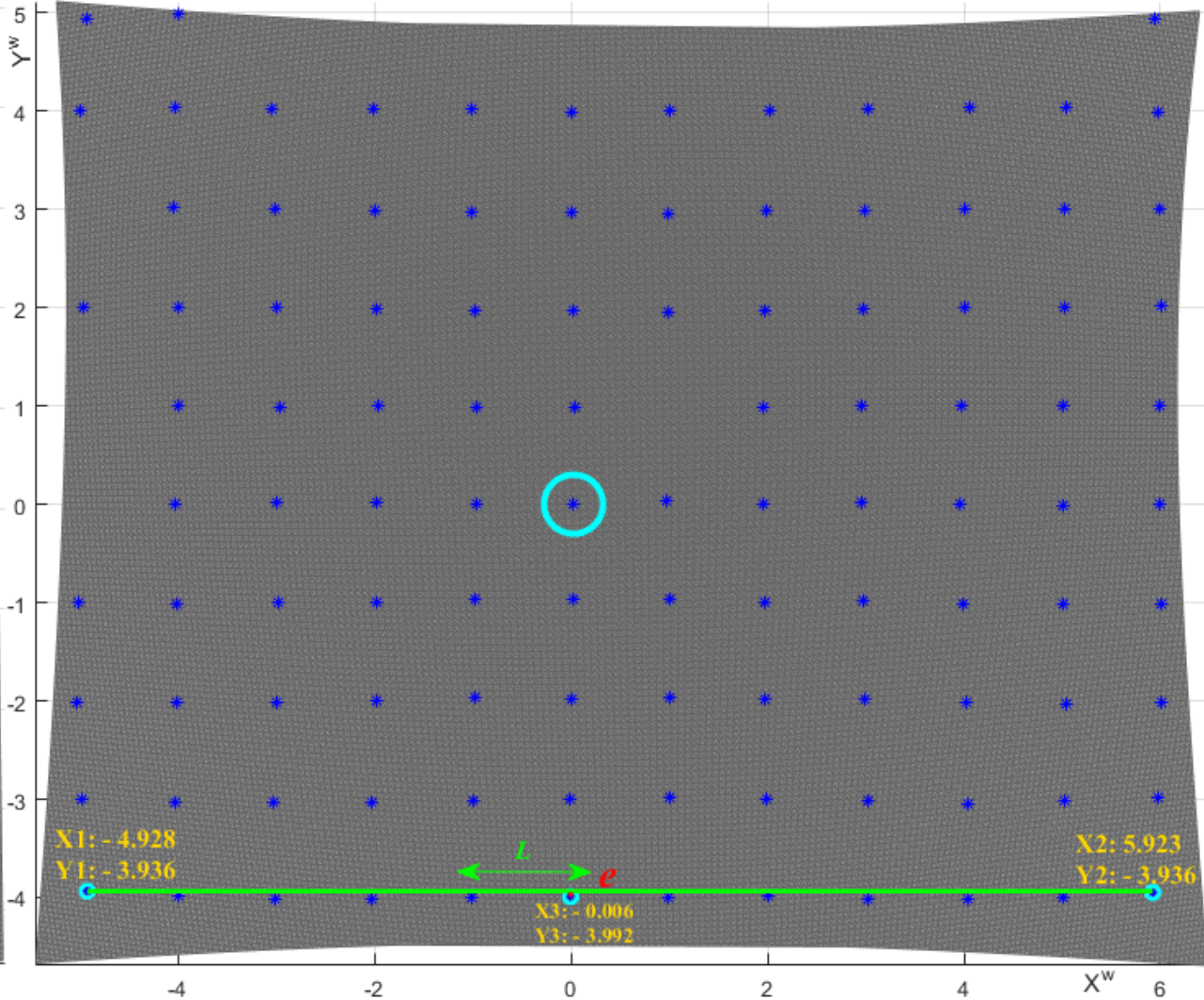
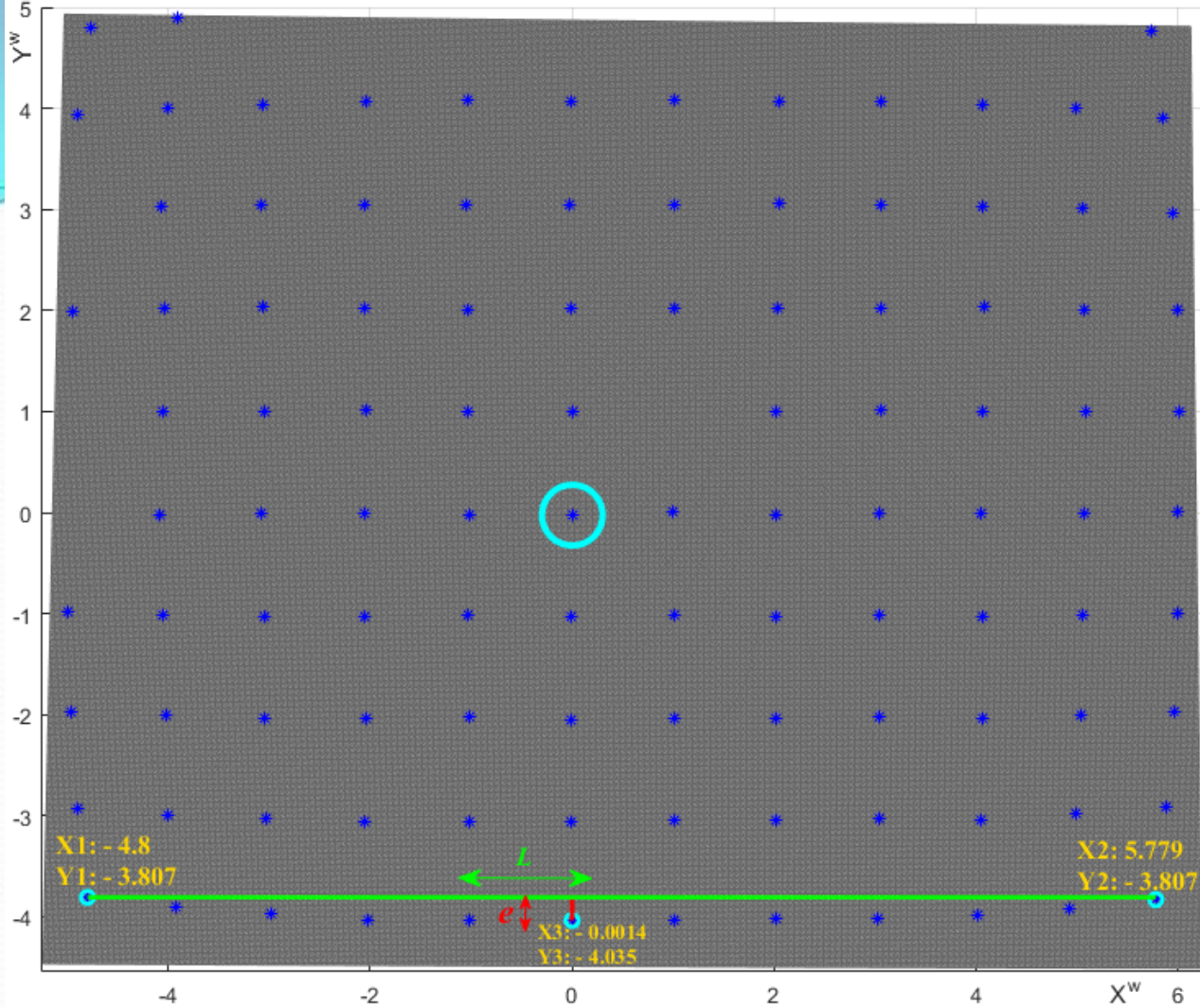
$$\begin{aligned} Y^W = & a_{21}C^4 + a_{22}C^3R + a_{23}C^2R^2 + a_{24}CR^3 + a_{25}R^4 + a_{26}C^3 + a_{27}C^2R \dots \\ & + a_{28}CR^2 + a_{29}R^3 + a_{210}C^2 + a_{211}CR + a_{212}R^2 + a_{213}C + a_{214}R + a_{215} \end{aligned}$$



$$d(\%) = e / L$$

$$d_0 = (R3 - R1) / (C2 - C1) = (403 - 393) / (492 - 20) = 2.1\%$$

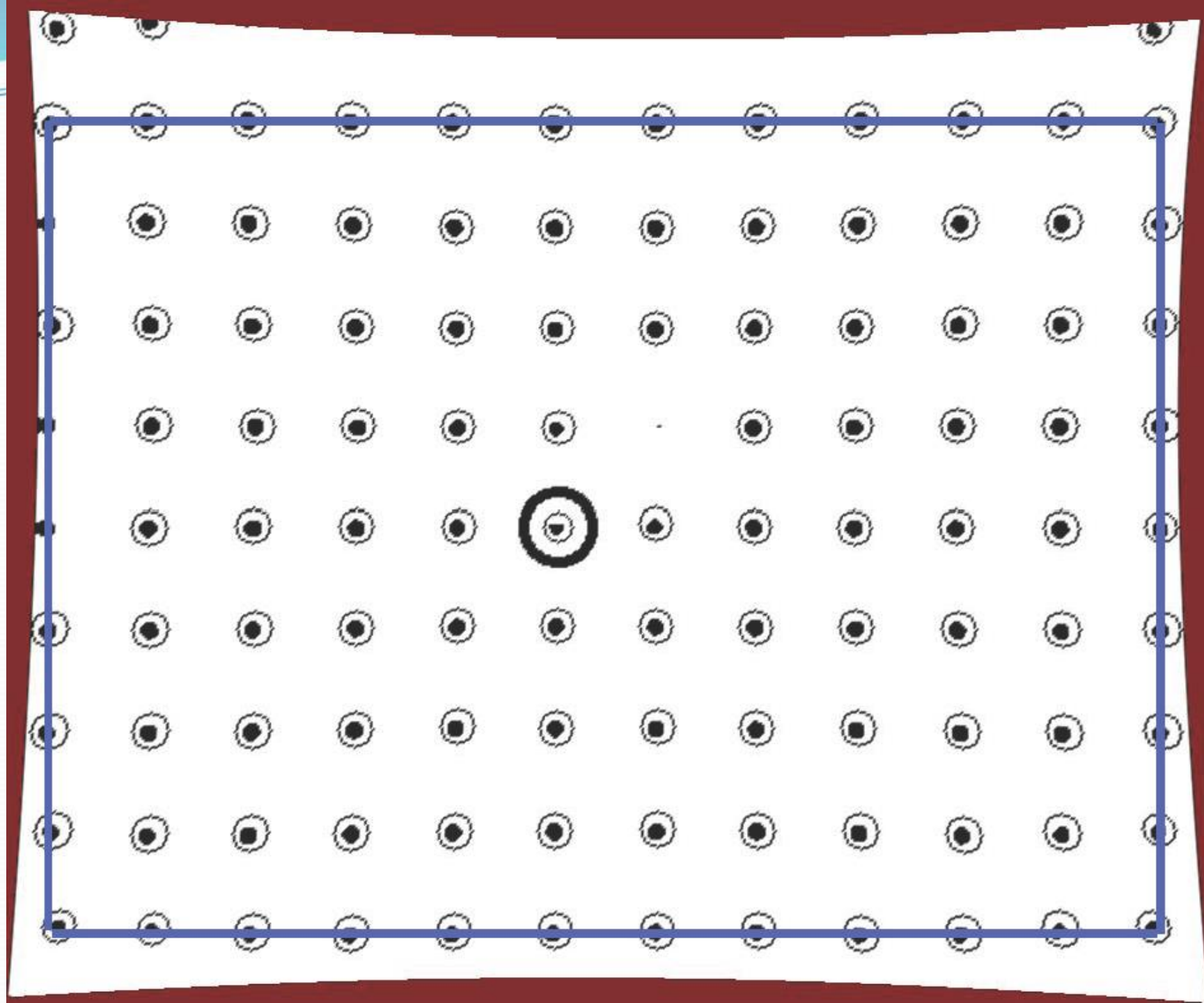
$$d_1 = (Y1 - Y3) / (X2 - X1) = [-3.772 + 4.004] / [5.713 + 4.735] = 2.2\%$$



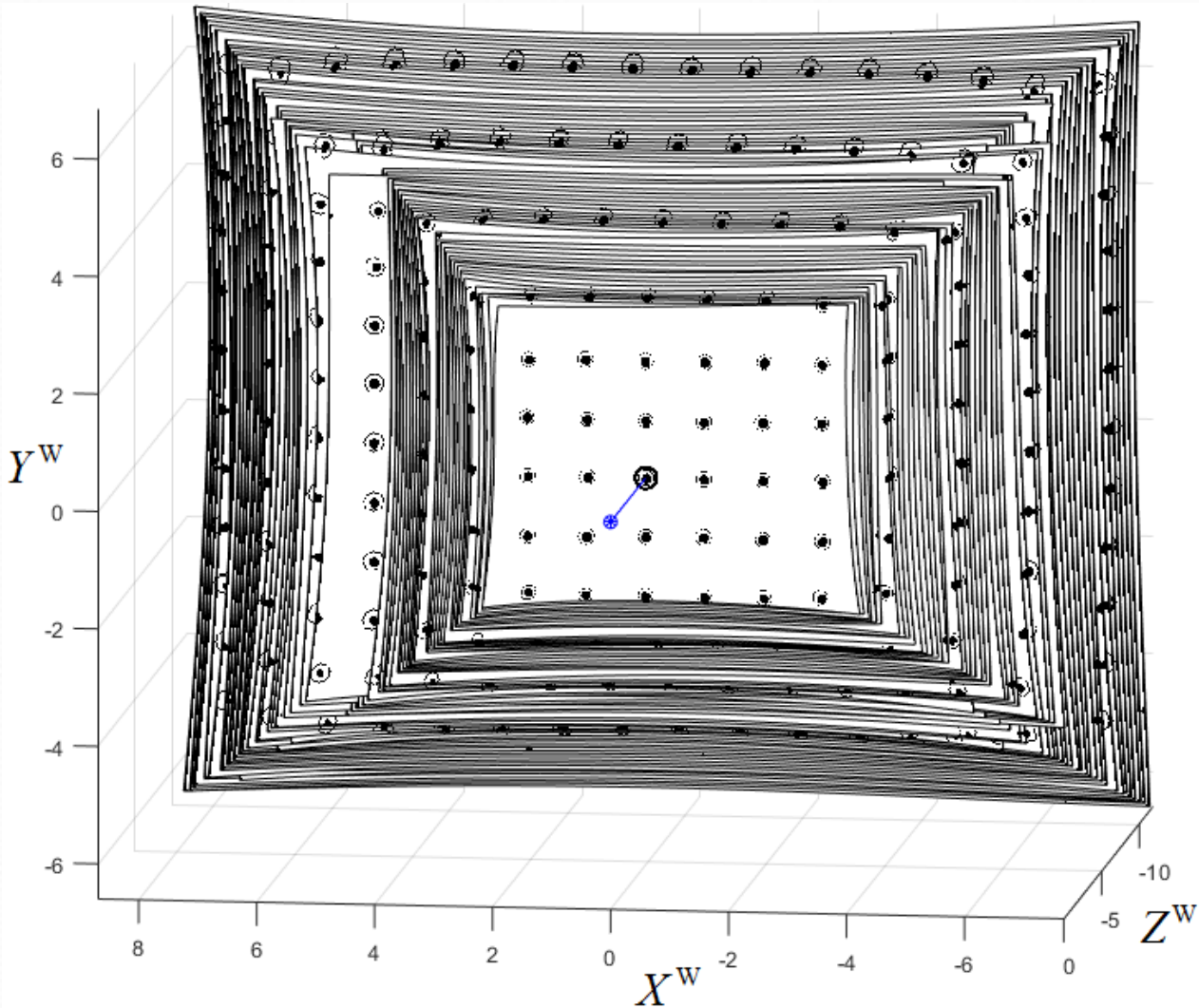
$$d(\%) = e / L$$

$$d_2 = [-3.807 - (-4.035)] = [5.779 - (-4.8)] = 2.1\%$$

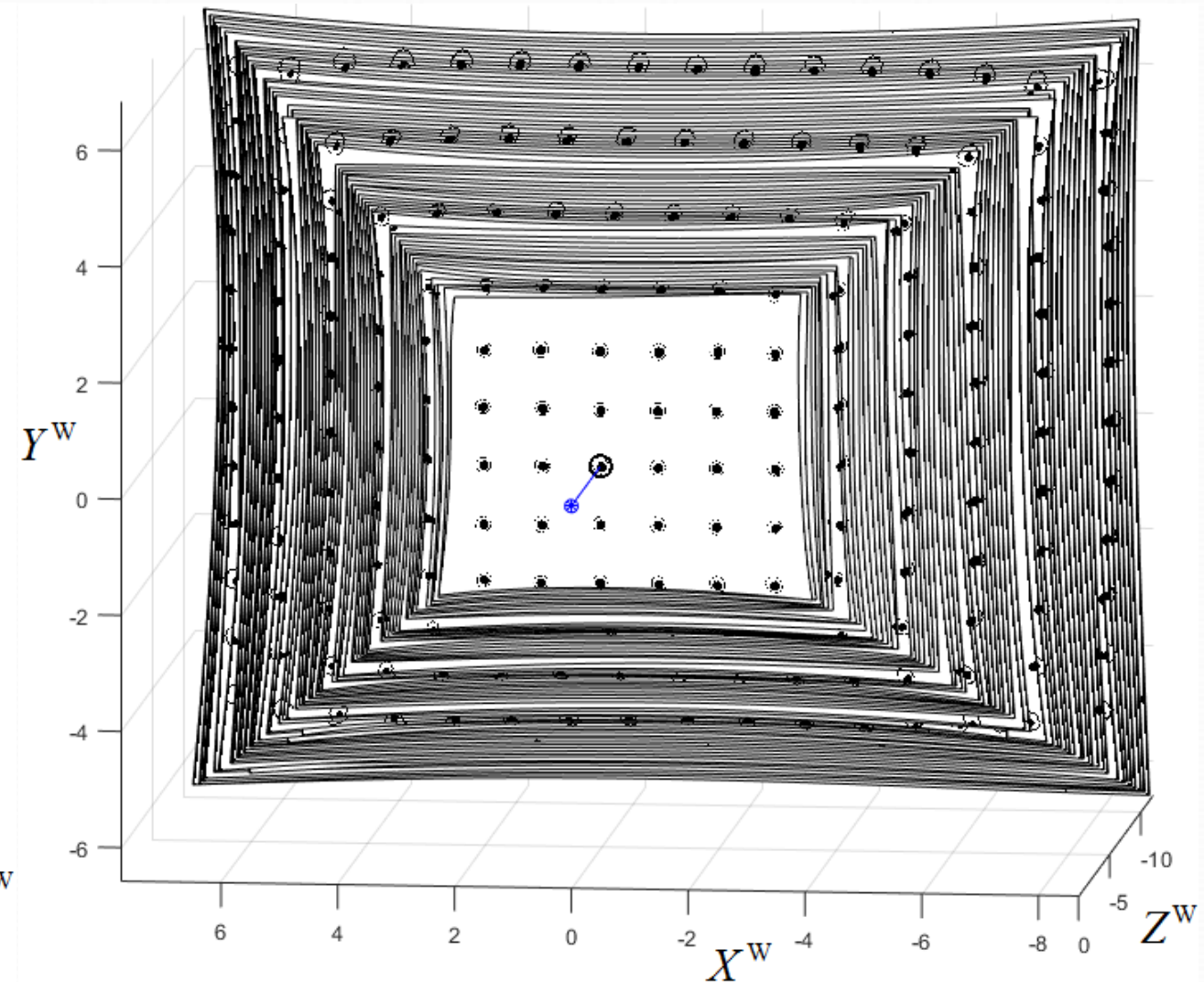
$$d_4 = [-3.936 - (-3.992)] = [5.923 - (-4.928)] = 0.516\%$$



63 frames from 1.165m to 2.565m, 25mm / frame

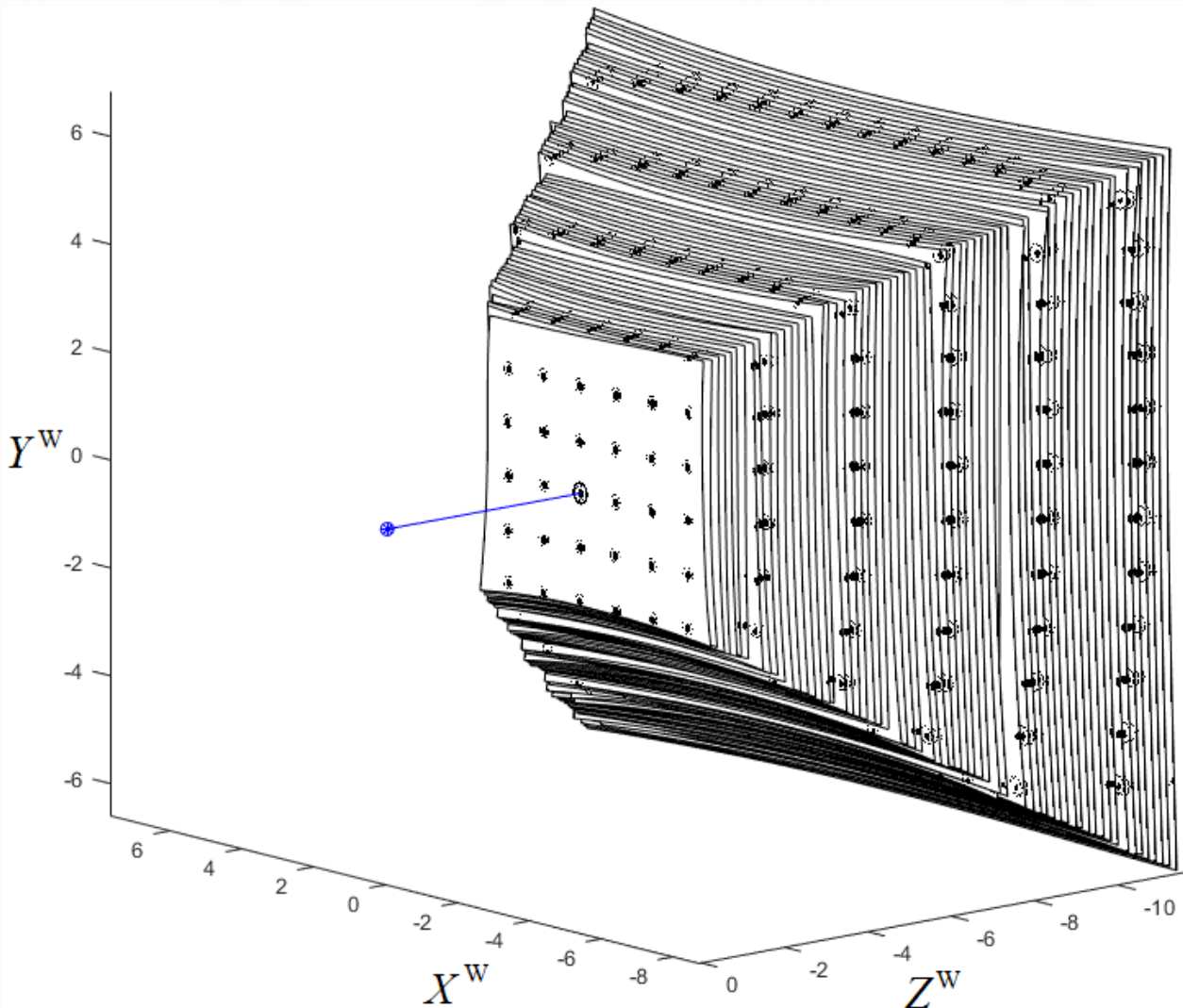


Staggered

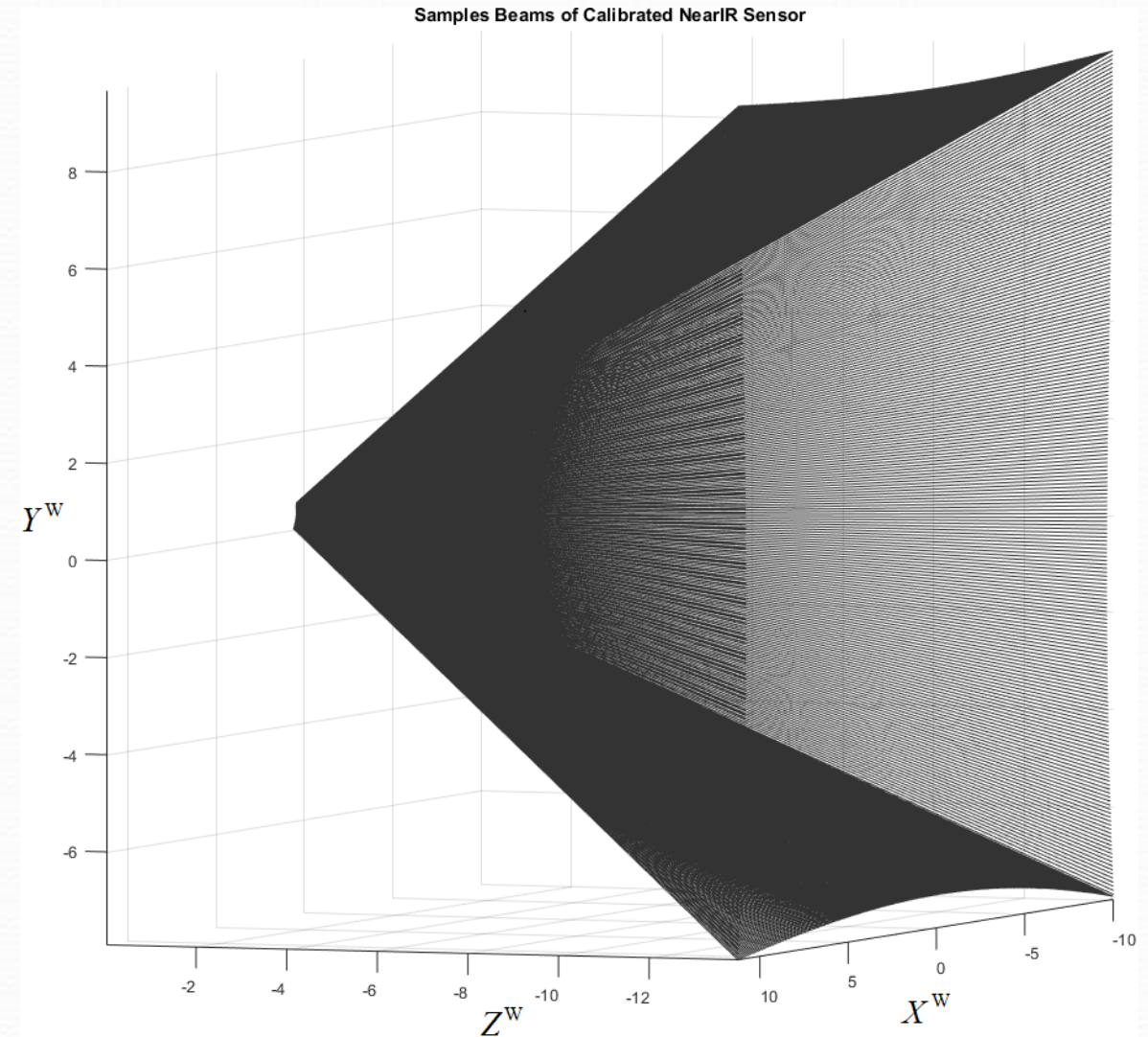


Unified

63 frames, generate Pixels' Beam Equations

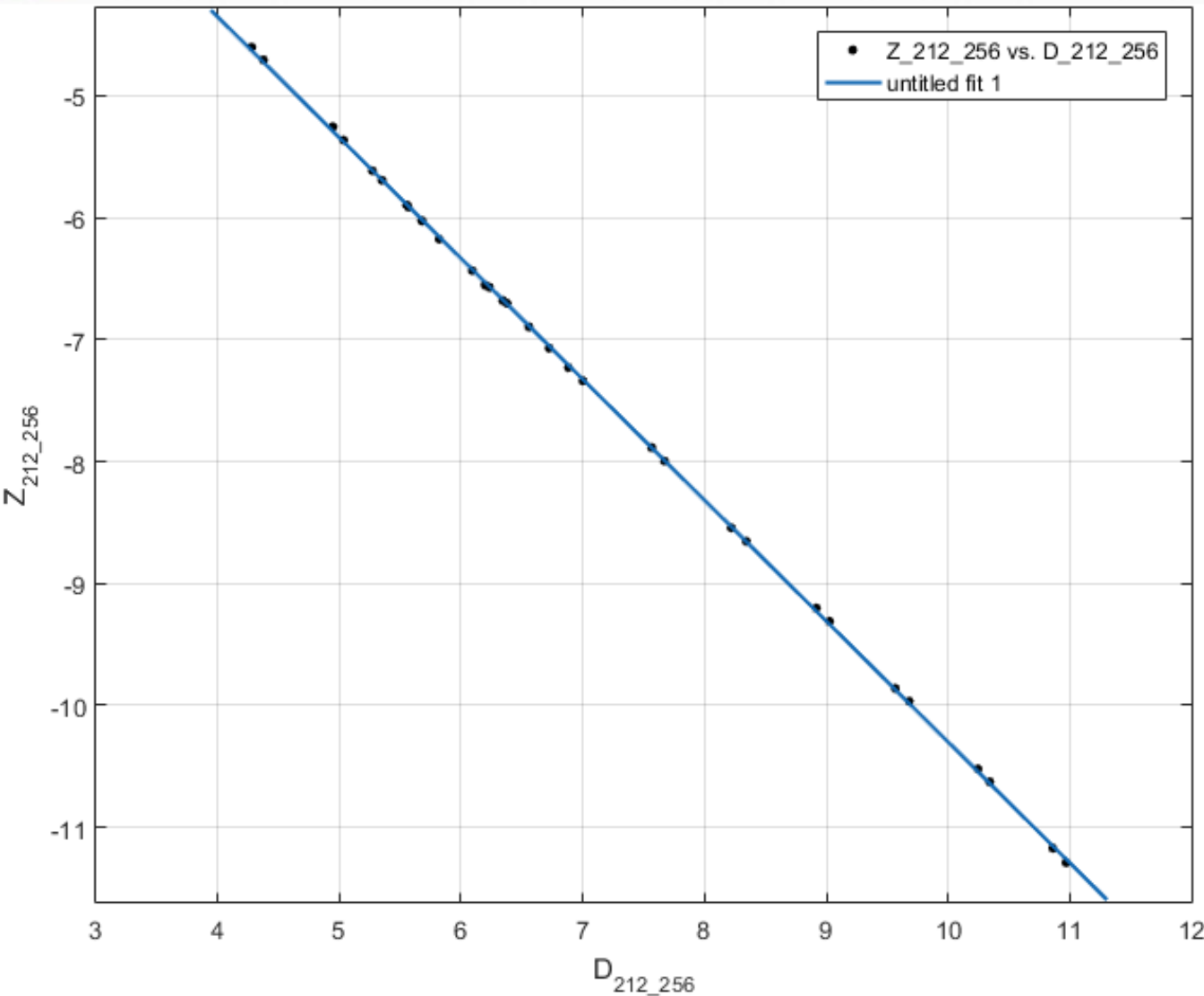


Frustum



Linear Z^w to $X^w Y^w$: Pixels' View

Per-Pixel D to Z^W mapping



D to Z^W Polynomial Fit

Data at pixel (212, 256)
from 32 frames



Linear in this example



$$Z^W[m, n] = e[m, n]D[m, n] + f[m, n]$$

Generate Look-Up Table

- Size: *Width - Height - 6* (512*424*6)
- Data: $X^W Y^W Z^W D$
- Pre-Process: (for every frame)
 - Find best-fit plane equation $D = aX^W + bY^W + c$
 - Throw away 10% pixels of worst D

- Mappings:

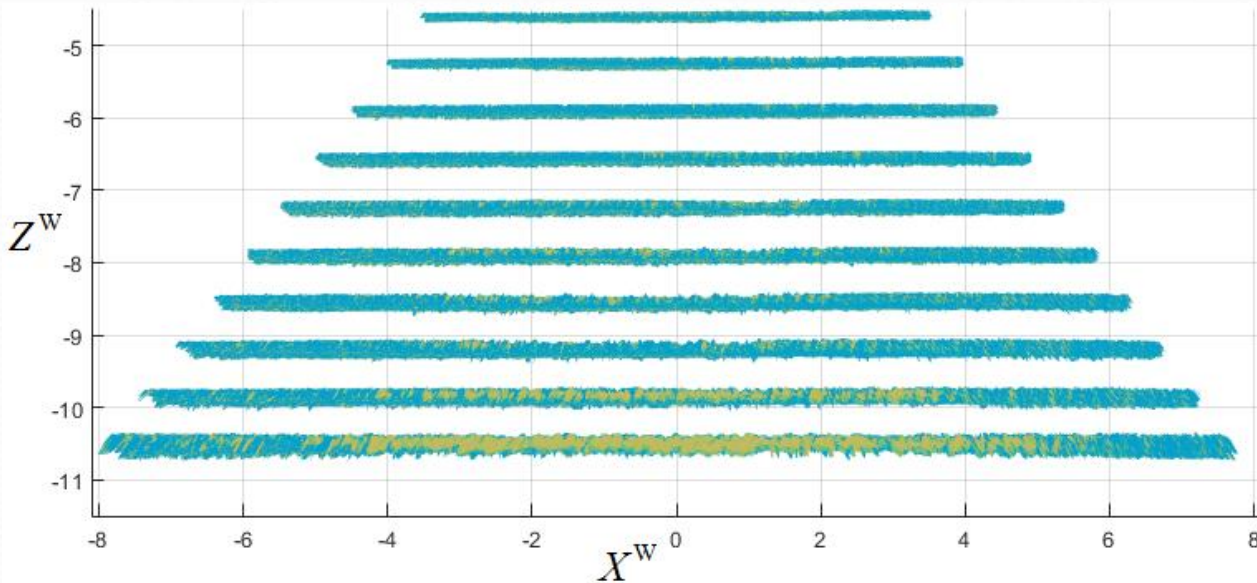
Fragment Shader: 3 per-pixel linear mappings, 6 parameters

$$X^W[m, n] = a[m, n]Z^W[m, n] + b[m, n]$$

$$Y^W[m, n] = c[m, n]Z^W[m, n] + d[m, n]$$

$$Z^W[m, n] = e[m, n]D[m, n] + f[m, n]$$

Depth Distortion Correction

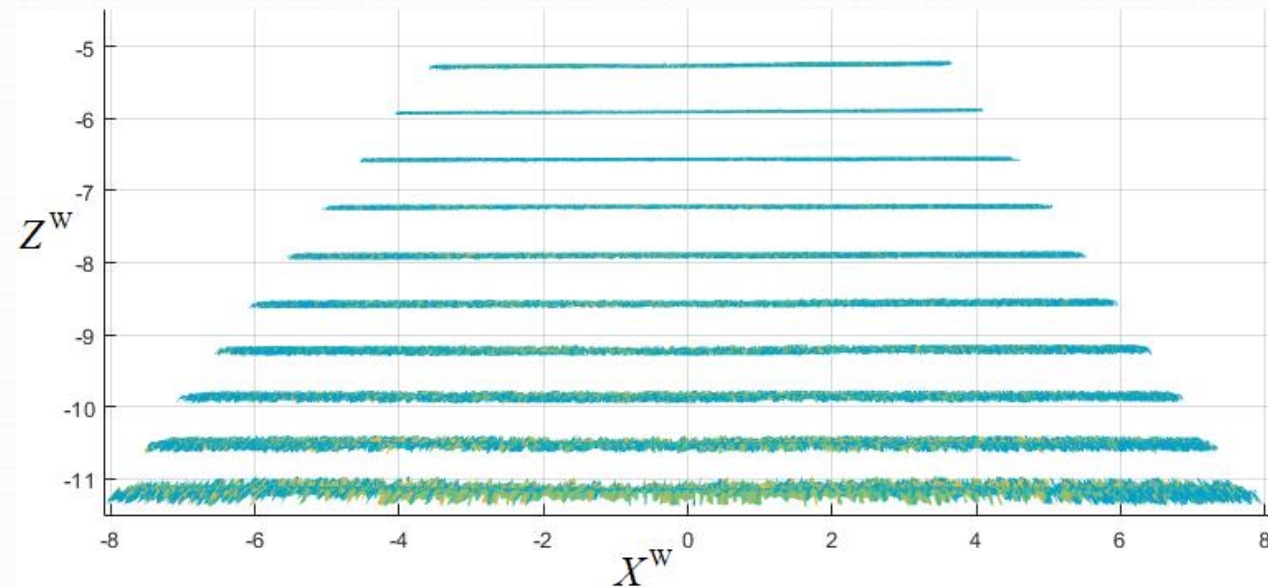


Raw Pin-Hole Reconstructions

Transformed
into World Space

By

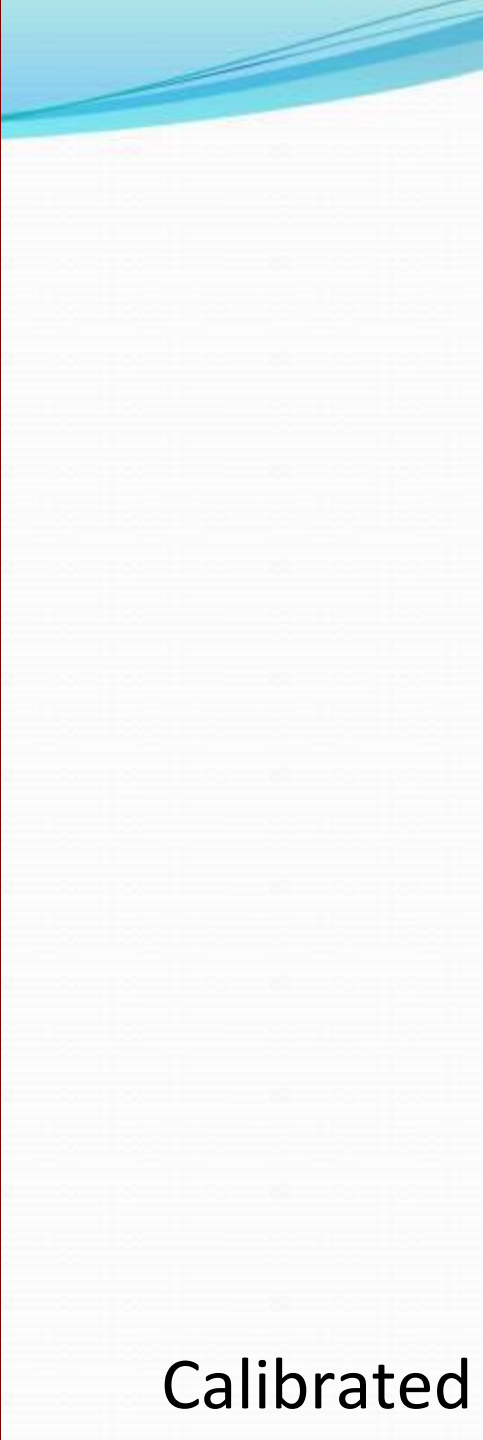
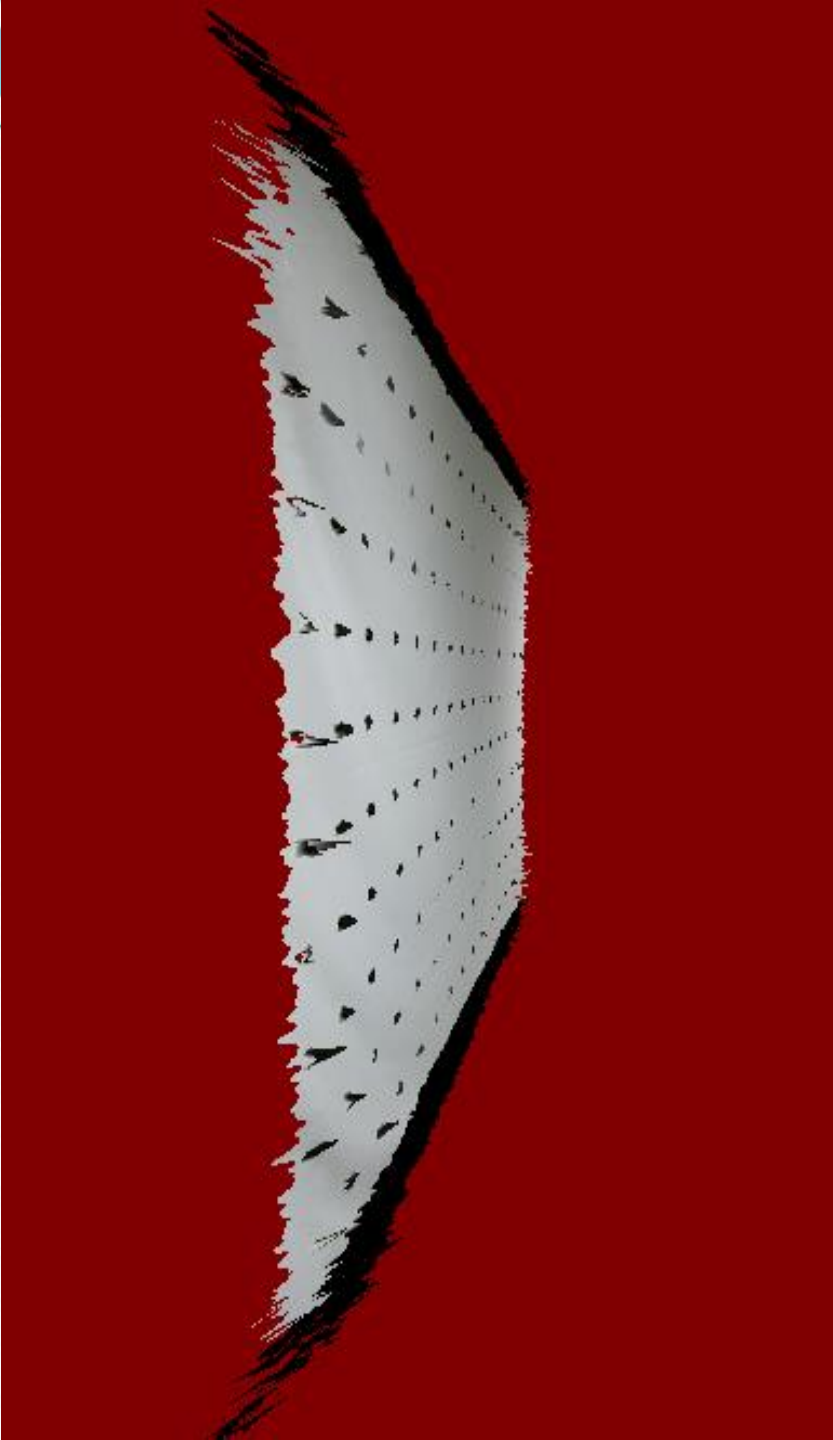
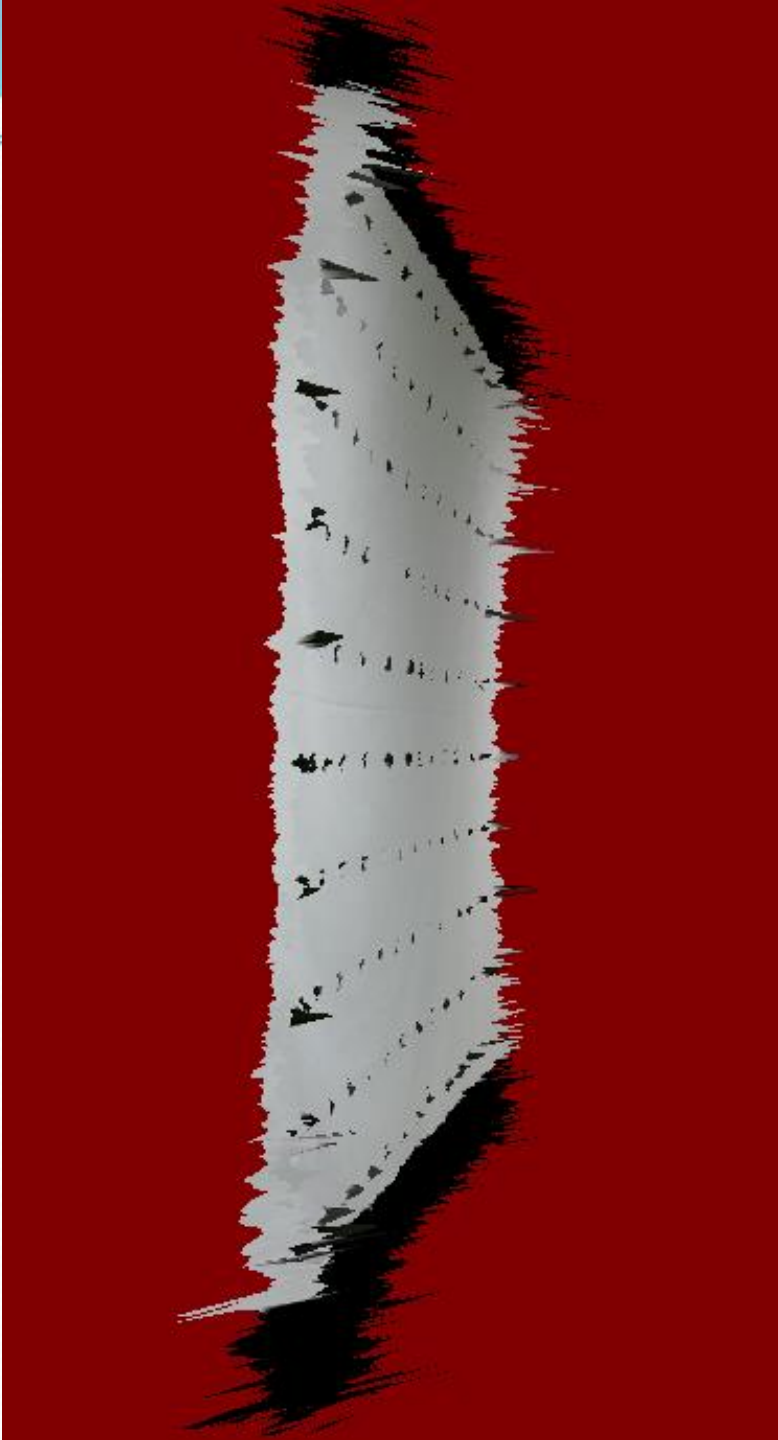
Best-Fit *Rotation and Translation*



Calibrated LUT Reconstructions

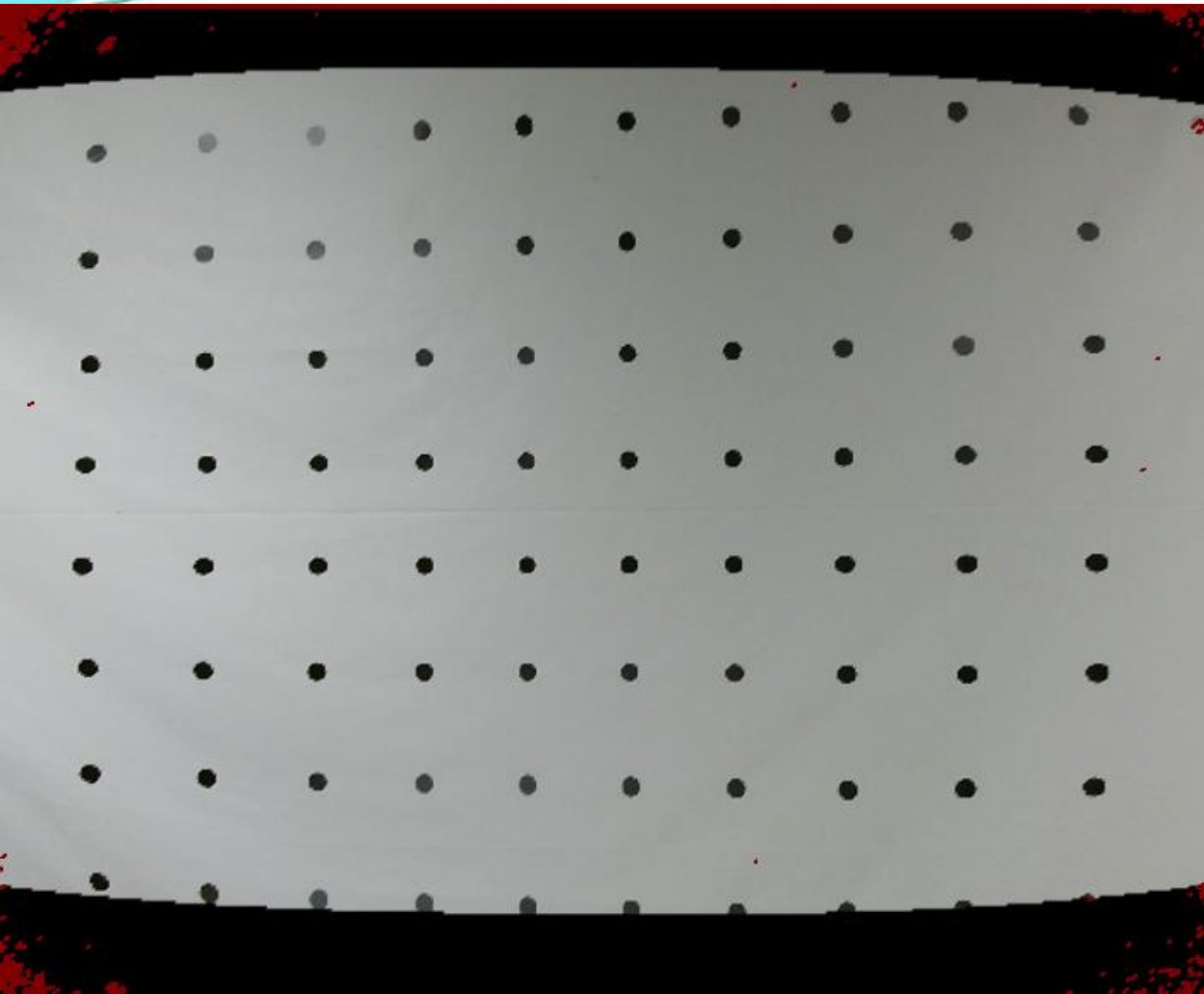


Before
Calibration

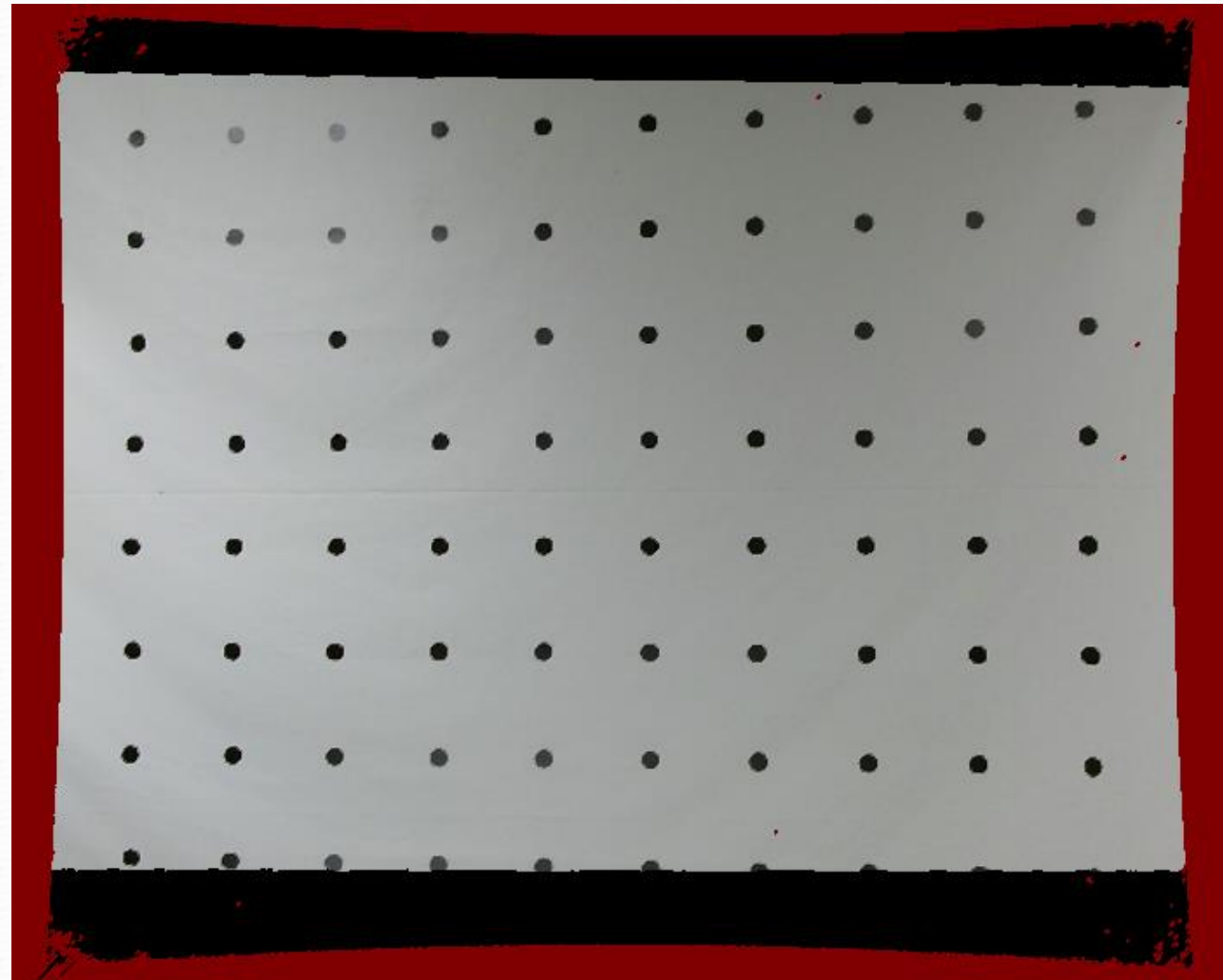


Calibrated

Lens Distortion Correction



Before Calibration



Calibrated

Conclusion

- Rail System

Infinite frames of data, dense calibration points, per-pixel D to Z^W mapping

- Data Collection (Per-Frame)

- Get Z^W from laser distance measurer ;

- Robust calibration points' extraction ;

Histogram Equalization, Adaptive Thresholding, Round Dots' Tracking

- Assign world space coordinates to calibration points ;

- Determine two-dimensional fourth order polynomial mapping ;

- Generate dense undistorted $X^W Y^W$.

- Pre-Process

- Unify staggered frames

- Throw away 10% noise pixels

- Generate LUT: *Width - Height - 6* (512*424*6)

- 3D Reconstruction: Undistorted 3D Reconstruction, No model, Only Textures

Future Works

- Hard Ware
 - Longer rail: segmented D to Z^W mapping in case a rough extrapolation;
 - Pattern Size and Distribution: based on resolution ;
 - 2D pattern to 3D pattern: in case NIR streams cannot be used, 3D pattern for depth streams analysis ;
 - Tracking module on rail: to substitute laser distance measure, such that it is possible to input Z^W record frames automatically.
- Software
 - Better DIP techniques ;
 - Higher order polynomial Z^W to $X^W Y^W$ mapping.



Questions?