

Министерство образования и науки Российской Федерации  
МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ  
(государственный университет)  
ФАКУЛЬТЕТ АЭРОФИЗИКИ И КОСМИЧЕСКИХ ИССЛЕДОВАНИЙ  
КАФЕДРА МЕХАНИКИ И ПРОЦЕССОВ УПРАВЛЕНИЯ

Определение углового движения по видеоизображению

Студент  
Julio Cesar Molina Saqui

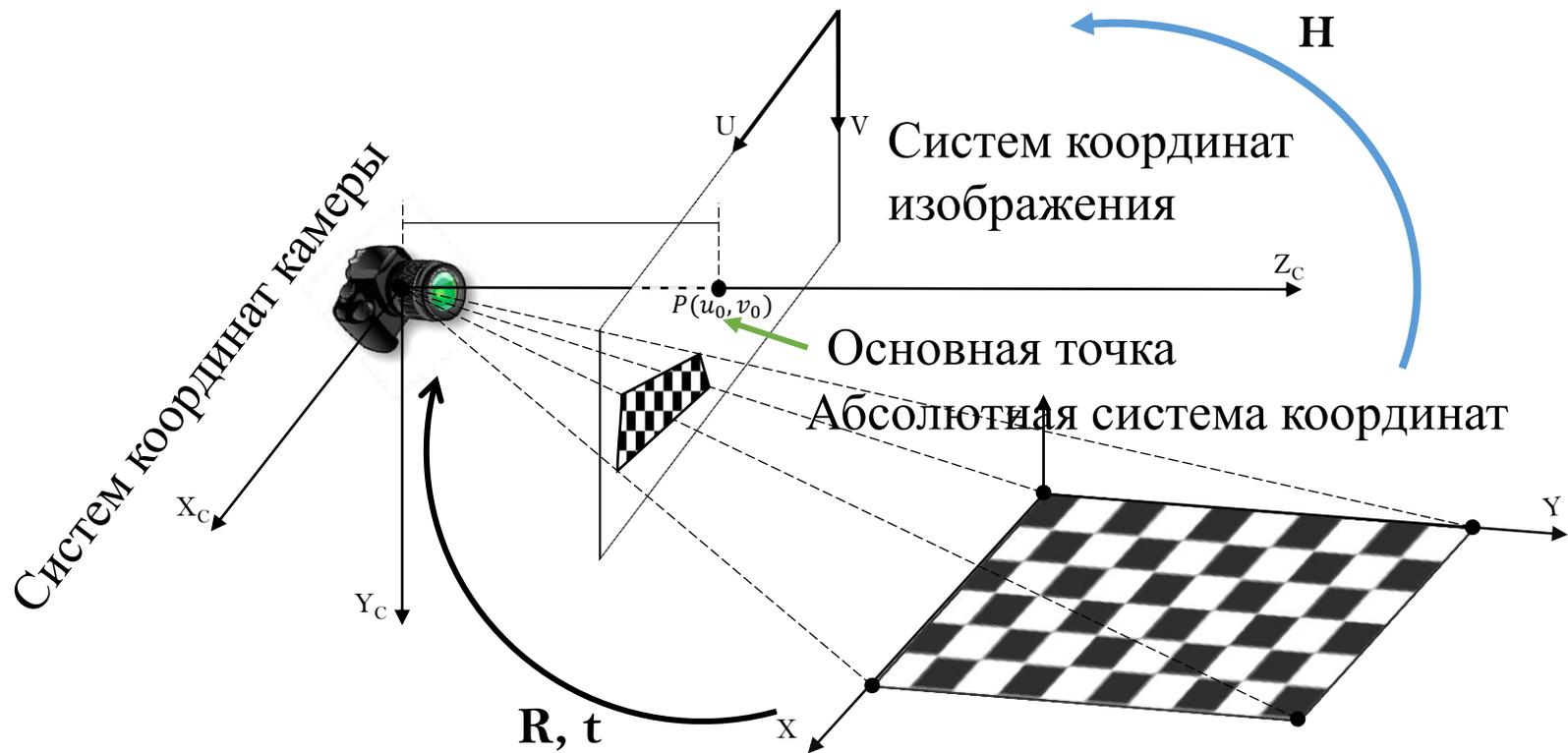
Научный руководитель  
*С.С. Ткачев*

г. Москва  
2018

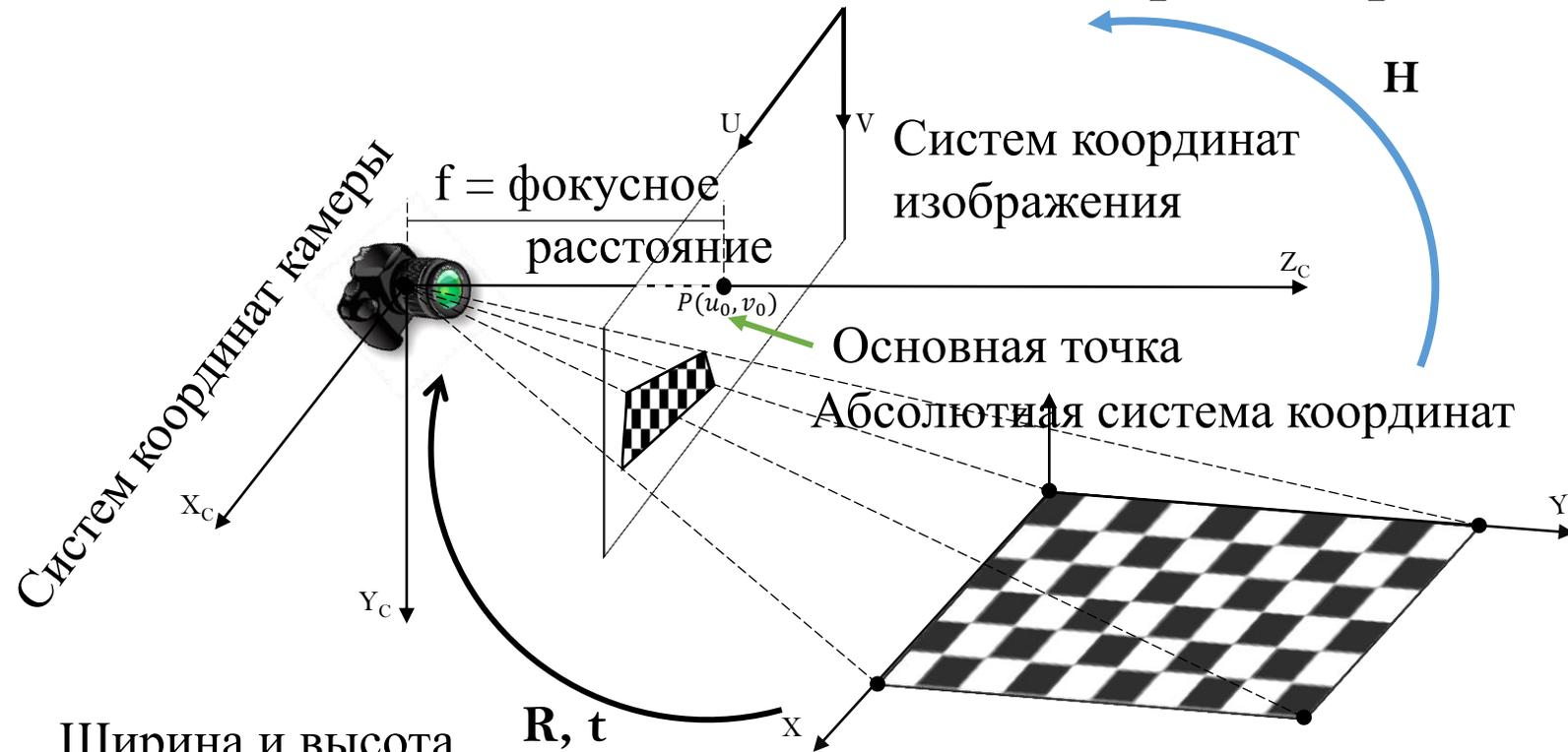
# Содержание

- **Введение**
- **Матрица перехода**
- **Алгоритм DLT**
- **Алгоритм для калибровки**
- **Опыт**

# Введение



# Матрица перехода



**Однородные координаты:**

2D точки  $(x, y)$  это изображение, которое можно представить как 3D вектор  $\tilde{P} = (\tilde{x}, \tilde{y}, \tilde{z})$ , на плоскости  $P^2$

Прямоугольные координаты  $P = (x, y)$   $\rightarrow$  Однородные координаты  $\tilde{P} = (x, y, 1)$

Однородные координаты  $\tilde{P} = (\tilde{x}, \tilde{y}, \tilde{z})$   $\rightarrow$  Прямоугольные координаты  $P = (x, y)$

$x = \frac{\tilde{x}}{\tilde{z}}, y = \frac{\tilde{y}}{\tilde{z}}$

Параметры внешней калибровки

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{bmatrix} = \underbrace{\begin{bmatrix} \alpha_x & s & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_K \underbrace{\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_C \underbrace{\begin{bmatrix} R & t \\ 0_{1 \times 3} & 1 \end{bmatrix}}_H \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{bmatrix} \approx \underbrace{\begin{bmatrix} h_1 & h_2 & h_3 & h_4 \\ h_5 & h_6 & h_7 & h_8 \\ h_9 & h_{10} & h_{11} & h_{12} \end{bmatrix}}_C \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

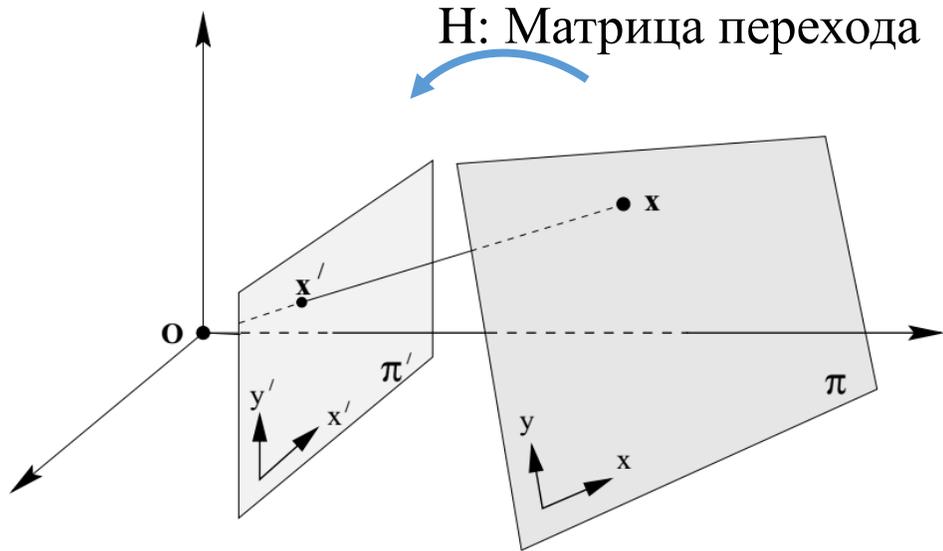
Параметры внутренней калибровки

С  $\leftarrow$  Матрица камеры

Однородные уравнения

Представление точки абсолютной системы координат в однородных координатах

# Алгоритм DLT (Direct Linear Transformation)



Прямоугольные координаты  $X'_i = (x'_i, y'_i)^T$   $\rightarrow$  Однородные координаты  $X'_i = (x'_i, y'_i, w'_i)^T$   
 $X_i = (x_i, y_i, z_i)^T$   $\rightarrow$   $X_i = (x_i, y_i, z_i, 1)^T$

Матрица перехода

$$H = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} = \begin{bmatrix} h^{1T} \\ h^{2T} \\ h^{3T} \end{bmatrix}$$

$$X'_i = \lambda H X_i \quad X'_i \times H X_i = 0 \quad \rightarrow$$

$$X'_i = \begin{bmatrix} h^{1T} \\ h^{2T} \\ h^{3T} \end{bmatrix} X_i = \begin{bmatrix} X_i h^{1T} \\ X_i h^{2T} \\ X_i h^{3T} \end{bmatrix}$$

$X'_i$  и  $H X_i$  не имеют тот же модуль, но имеют тот же направление

$$X'_i \times \begin{bmatrix} X_i h^{1T} \\ X_i h^{2T} \\ X_i h^{3T} \end{bmatrix} = \begin{bmatrix} x'_i \\ y'_i \\ w'_i \end{bmatrix} \times \begin{bmatrix} h^{1T} X_i \\ h^{2T} X_i \\ h^{3T} X_i \end{bmatrix} = \begin{bmatrix} y'_i h^{3T} X_i - w'_i h^{2T} X_i \\ w'_i h^{1T} X_i - x'_i h^{3T} X_i \\ x'_i h^{2T} X_i - y'_i h^{1T} X_i \end{bmatrix} \xrightarrow{h^{jT} X_i = X_i^T h^j} \begin{bmatrix} 0^T & -w'_i X_i^T & y'_i X_i^T \\ w'_i X_i^T & 0^T & -x'_i X_i^T \\ -y'_i X_i^T & -x'_i X_i^T & 0^T \end{bmatrix} \begin{bmatrix} h^1 \\ h^2 \\ h^3 \end{bmatrix} = 0$$

- Простой линейный алгоритм.
- Необходимо найти 4 соответствия

Хотя тут три уравнения, только два из них линейно независимы.

$$\begin{bmatrix} 0^T & -w'_i X_i^T & y'_i X_i^T \\ w'_i X_i^T & 0^T & -x'_i X_i^T \\ -y'_i X_i^T & -x'_i X_i^T & 0^T \end{bmatrix} \begin{bmatrix} h^1 \\ h^2 \\ h^3 \end{bmatrix} = 0$$

$A_i h = 0$ , решается с помощью SVD

$$h = [h_1 h_2 h_3 h_4 h_5 h_6 h_7 h_8 h_9]^T$$

# Оценка матрицы перехода H

Однородные координаты

$$\underbrace{\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix}}_{X'} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \underbrace{\begin{bmatrix} x \\ y \\ z \end{bmatrix}}_X$$

$$\begin{aligned} x' &= h_1x + h_2y + h_3z \\ y' &= h_4x + h_5y + h_6z \\ w' &= h_7x + h_8y + h_9z \end{aligned}$$



$$\tilde{X}' = (\tilde{x}', \tilde{y}')^T = \left( \frac{x'}{w'}, \frac{y'}{w'} \right)^T$$

Неоднородные координаты

$$\begin{cases} \tilde{x}' = \frac{h_1x + h_2y + h_3z}{h_7x + h_8y + h_9z} \\ \tilde{y}' = \frac{h_4x + h_5y + h_6z}{h_7x + h_8y + h_9z} \end{cases}$$

$$\tilde{X}' = f(h); \text{ где } h = [h_1 h_2 h_3 h_4 h_5 h_6 h_7 h_8 h_9]^T$$

Методом Ньютона:

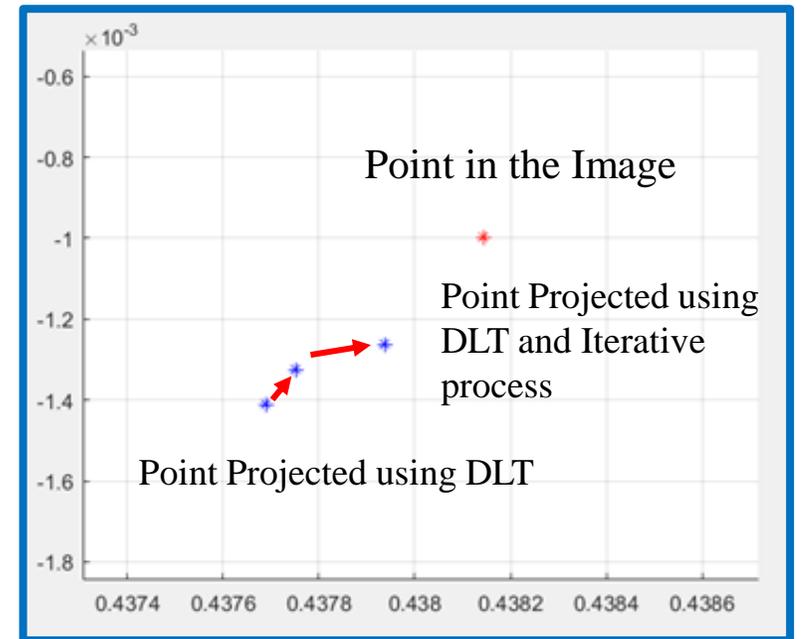
$$\epsilon_n = f(h_n) - X \quad h_0, \epsilon_0 = f(h_0) - X$$

$$h_{n+1} = h_n + \Delta h, \text{ где } \Delta h = (J^T J)^{-1} J^T \epsilon_n, \text{ and } J = \frac{\partial \tilde{X}}{\partial h}$$

Матрица Якоби (J) :

$$\frac{\partial \tilde{X}}{\partial h} = \begin{bmatrix} \frac{\partial \tilde{x}}{\partial h_1} & \frac{\partial \tilde{x}}{\partial h_2} & \dots & \frac{\partial \tilde{x}}{\partial h_9} \\ \frac{\partial \tilde{y}}{\partial h_1} & \frac{\partial \tilde{y}}{\partial h_2} & \dots & \frac{\partial \tilde{y}}{\partial h_9} \end{bmatrix} = \begin{bmatrix} x & y & z & 0 & 0 & 0 & -\tilde{x}'x & -\tilde{x}'y & \tilde{x}'z \\ 0 & 0 & 0 & x & y & z & -\tilde{y}'x & -\tilde{y}'y & \tilde{y}'z \end{bmatrix}$$

$$\frac{\partial \tilde{X}}{\partial h} = \begin{bmatrix} X^T & 0^T & -\tilde{x}'X^T \\ 0^T & X^T & -\tilde{y}'X^T \end{bmatrix}$$



Итерационный метод улучшает результат при проектировании точек от WCS до системы координат изображения

# Алгоритм для калибровки

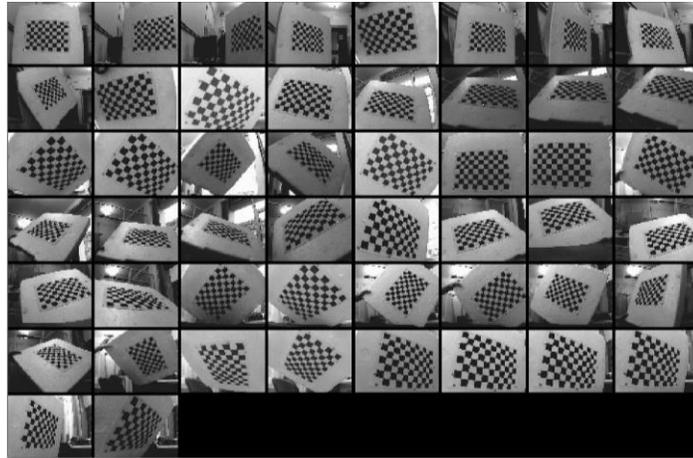
Получение соответствия для калибровки



Вычисление матрицы перехода

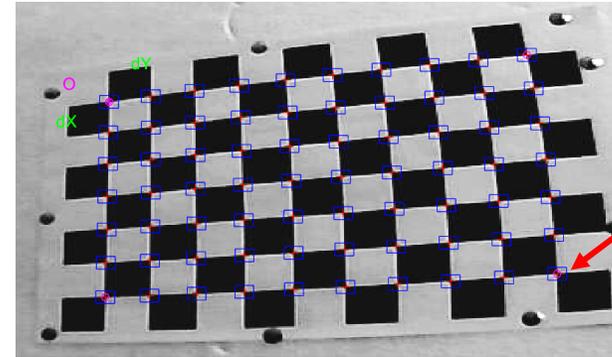


Запуск параметров внутреннего калибровки



50 изображения для калибровки

точки изображения  $k$ .  
где  $k = 1, 2, \dots, 50$



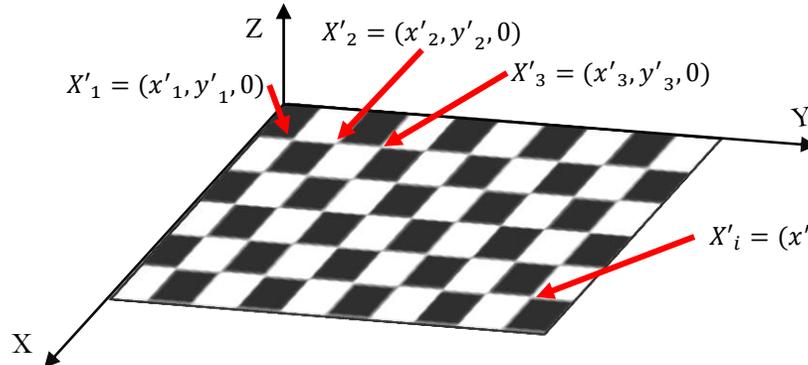
$$X'_i = (x'_i, y'_i)$$

Получается 70 точек в однородных координатах за изображение

Прямоугольные координаты  $X'_i = (x'_i, y'_i)^T$   $\rightarrow$  Однородные координаты  $X'_i = (x'_i, y'_i, 1)^T$

$$P'_k = [X'_1 \quad X'_2 \quad \dots \quad X'_i] = \begin{bmatrix} x'_1 & x'_2 & \dots & x'_i \\ y'_1 & y'_2 & \dots & y'_i \\ 1 & 1 & \dots & 1 \end{bmatrix}$$

Инерциальная система координат



Прямоугольные координаты  $X_i = (x_i, y_i, z_i)^T$   $\rightarrow$  Однородные координаты  $X_i = (x_i, y_i, z_i, 1)^T$

$$X_i = (x_i, y_i, z_i)^T \rightarrow X_i = (x_i, y_i, z_i, 1)^T$$

$X_i = (x_i, y_i, 1)^T$ , because of  $z_i = 0$

$$P = [X_1 \quad X_2 \quad \dots \quad X_i] = \begin{bmatrix} x_1 & x_2 & \dots & x_i \\ y_1 & y_2 & \dots & y_i \\ 1 & 1 & \dots & 1 \end{bmatrix}$$

70 точек в однородных координатах

# Алгоритм для калибровки

Получение соответствия для калибровки

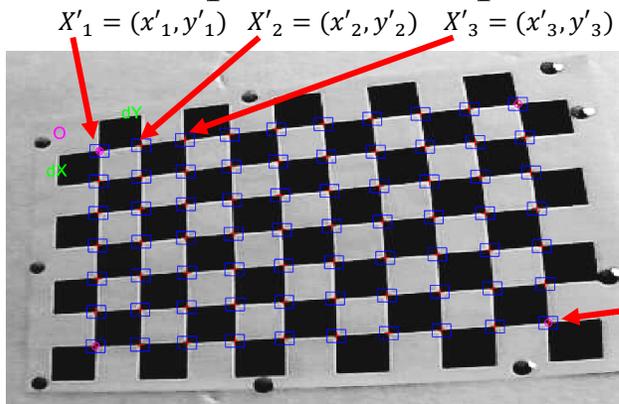


Вычисление матрицы перехода



Запуск параметров внутреннего калибровки

Систем координат изображения



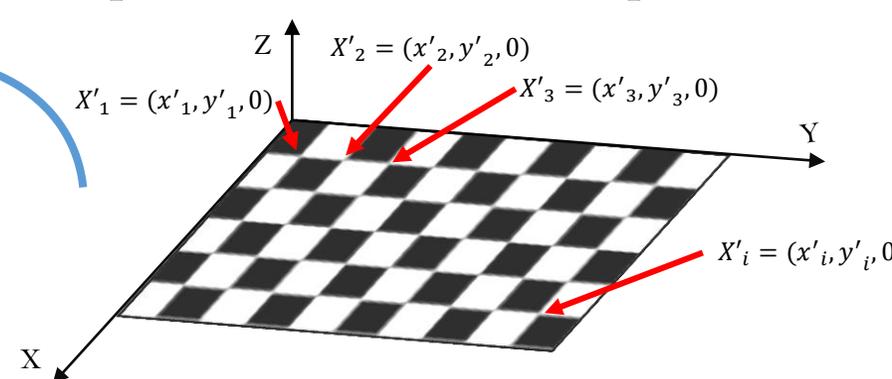
$$P'_k = [X'_1 \quad X'_2 \quad \dots \quad X'_i]$$

$$H_{prec}^k = \begin{bmatrix} \beta_x & 0 & -\beta_x \bar{x} \\ 0 & \beta_y & -\beta_y \bar{y} \\ 0 & 0 & 1 \end{bmatrix}$$

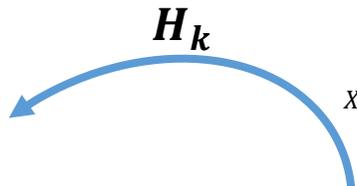
$$P1'_k = H_{prec}^k P'_k \longrightarrow$$

Данный процесс повторяется, чтобы получить  $H_1, H_2, \dots, H_k$   
 $k$  : число изображений

Инерциальная система координат



$$P = [X_1 \quad X_2 \quad \dots \quad X_i]$$



$$h/h_9 \longrightarrow H = (H_{prec}^k)^{-1} [h]_{3 \times 3}$$



$H_k$

# Алгоритм для калибровки

Получение соответствия для калибровки



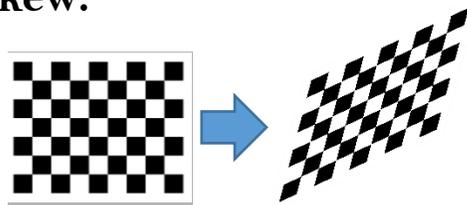
Вычисление матрицы перехода



Запуск параметров внутреннего калибровки



Skew:



$s = 0$ ; В современных камерах составляет почти 0

**Основная точка:** Назначается в центре изображения  
 $P(u_0, v_0) = (W/2, H/2)$

**Фокусное расстояние:** Определяется с использованием всех матриц перехода



$$\begin{bmatrix} f_x & S & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$f_x, f_y$ : фокусное расстояние

# Алгоритм для калибровки

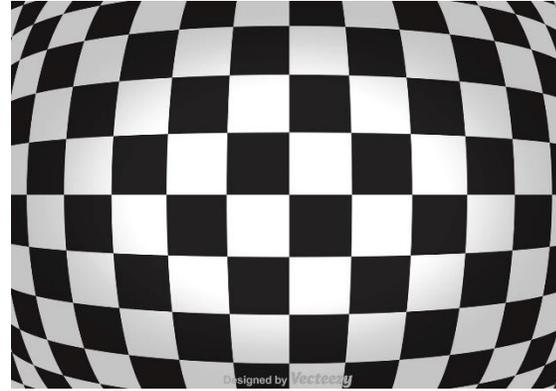
Получение соответствия для калибровки



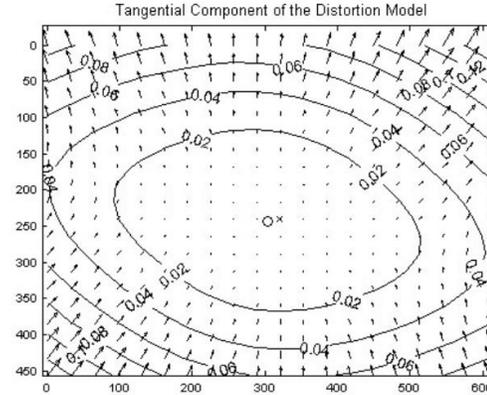
Вычисление матрицы перехода



Запуск параметров внутреннего калибровки



Радиальное Искажение



Тангенциальное Искажение

Итерационный метод компенсации

$$r = ((x'_i - u_0) / \alpha_x f, (y'_i - v_0) / \alpha_y f)$$

$$r_x = r_x - s r_y \quad \text{Undo skew}$$

$$Radial = 1 + k_1 r^2 + k_2 r^4 + k_3 r^6 \quad Tangential = (2p_1 r_x r_y + p_2 (r^2 + 2r_x^2), p_1 (r^2 + 2r_y^2) + 2p_2 r_x r_y)$$

$$X_{-U_i} = \begin{bmatrix} x u_i \\ x u_i \end{bmatrix} = (r - Tangential) / Radial$$

Точка переводится с системы координат изображения в систему координат камеры

Скомпенсированные точки от радиального и тангенциального искажения

Distortion coefficients ( $K_D$ ):  $K_D = [k_1 \quad k_2 \quad p_1 \quad p_2 \quad k_3]$  Запуск  $K_D = [0 \ 0 \ 0 \ 0 \ 0]$

# Алгоритм для калибровки



Неискаженные точки изображения  $K$   
в системе координат камеры  
 $U'_k$

Точки объекта в инерциальной  
системе координат  
 $P = [X_1 \ X_2 \ \dots \ X_i]$

Вычисление  
матрицы  
перехода

Матрица перехода  
с инерциальной системы  
координат в систему  
координат камеры

$$H_k = [R|t]$$

Вектор переноса относительно CCS  
Матрица поворота относительно CCS

## Алгоритм для калибровки

Процесс основан на итеративном процессе, использующем проекцию всех точек в WCS на систему координат изображения используя модель измерения, инициализированные внутренние параметры и итеративный метод для уточнения оценки "R" и "t".

Запуск параметров внешнего калибровки

Итеративный процесс улучшения оценки параметров внешней калибровки

Главный процесс оптимизации

$$X_c = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = RX_i + t \rightarrow X_{cp} = \begin{bmatrix} x_c/z_c \\ y_c/z_c \end{bmatrix} = \begin{bmatrix} x_{cp} \\ y_{cp} \end{bmatrix} \quad \text{Проекция координат с WCS на CCS}$$

$$r^2 = x_{cp}^2 + y_{cp}^2 \quad \text{Радиальное искажение} \quad \text{Тангенциальное искажение}$$

$$X_d = \begin{bmatrix} x_d \\ y_d \end{bmatrix} = \begin{bmatrix} x_{cp}(1 + k_1r^2 + k_2r^4 + k_3r^6) + 2p_1x_{cp}y_{cp} + p_2(r^2 + 2x_{cp}^2) \\ y_{cp}(1 + k_1r^2 + k_2r^4 + k_3r^6) + 2p_2x_{cp}y_{cp} + p_1(r^2 + 2y_{cp}^2) \end{bmatrix}$$

$$Xp_i = \begin{bmatrix} x_p \\ y_p \end{bmatrix} = \begin{bmatrix} (x_d + sy_d)f + u_0 \\ (x_d)f + v_0 \end{bmatrix}$$

$Xp_i$ : Точка, уже проецируемая в систему координат изображения из системы мировых координат

$$E_k = [X'_1 - Xp_1 \quad X'_2 - Xp_2 \quad \dots \quad X'_i - Xp_i]$$

матрицы Якоби(J) :

$$J = \begin{bmatrix} \frac{\partial Xp_1}{\partial R} & \frac{\partial Xp_1}{\partial t} \\ \frac{\partial Xp_2}{\partial R} & \frac{\partial Xp_2}{\partial t} \\ \frac{\partial Xp_i}{\partial R} & \frac{\partial Xp_i}{\partial t} \end{bmatrix}$$

Из-за нелинейности модели камеры одновременная оценка параметров включает применение итерационного алгоритма

# Алгоритм для калибровки

В этом процессе оптимизации обновляются все параметры внутренней и внешней калибровки обновляются.

$$F = [f_x \quad f_y]$$

$$C = [u_0 \quad v_0]$$

$$K_D = [k_1 \quad k_2 \quad p_1 \quad p_2 \quad k_3]$$

$$M = [F \quad C \quad S \quad K_D \quad R_1 \quad t_1 \quad R_2 \quad t_2 \quad \dots \quad R_k \quad t_k]$$

M : Глобальный вектор параметров

k: Количество изображений,

Внутренние параметры: Model FI8918W, 480x640 pixels

Фокусное расстояние:

$$f_x = 635.61266$$

$$f_y = 636.45016$$

Основная точка

$$u_0 = 320.88665$$

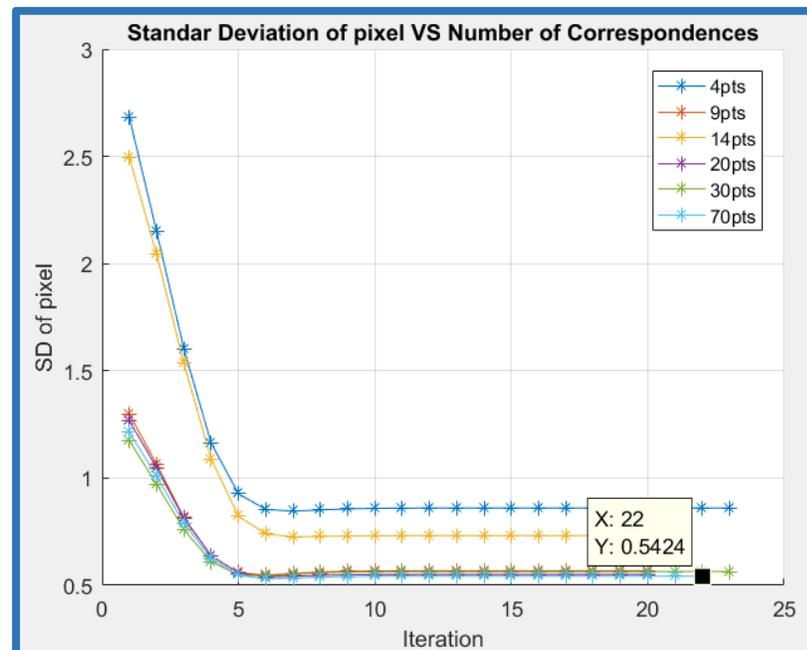
$$v_0 = 246.79063$$

Skew:  $S = 0.0$

искажение :  $K_1 = -0.44492$ ,  $K_2 = 0.25395$ ,

$$p_1 = 0.00246, p_2 = -0.00378, K_3 = 0.000$$

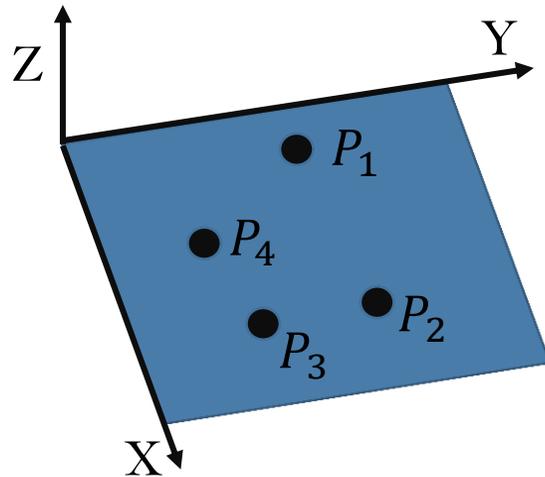
SD : 0.54 pixel



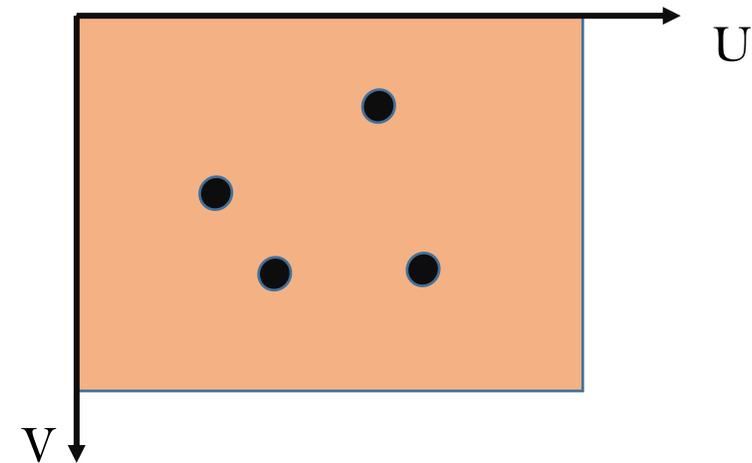
# Опыт

Важным шагом является определение координат объекта в инерциальной системе координат. Данные координаты должны быть обнаружены камерой и иметь возможность установить соответствие между точками из WCS и системой координат изображения

Инерциальная система координат



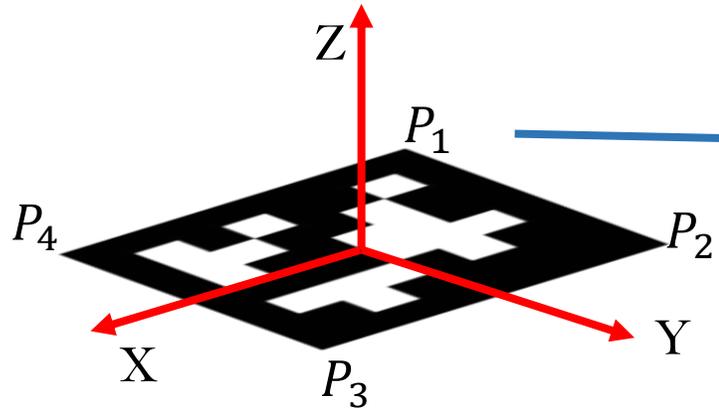
Систем координат изображения



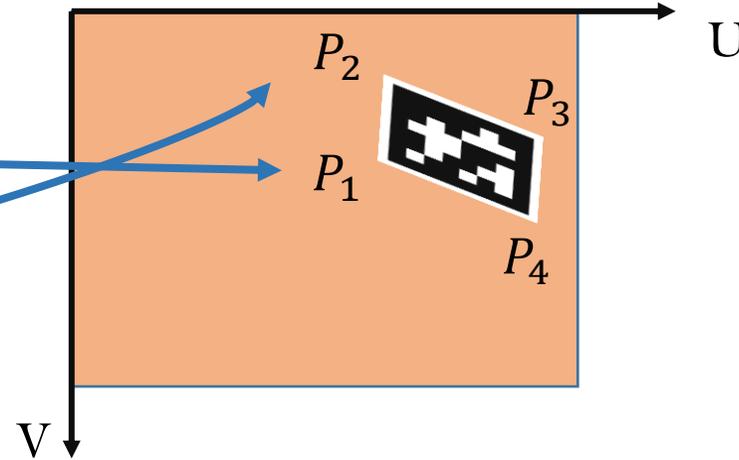
Как только эти четыре точки обнаружены и их положение в изображении оценивается, невозможно определить какая точка является P1, P2, P3 или P4

# Опыт

Инерциальная система координат



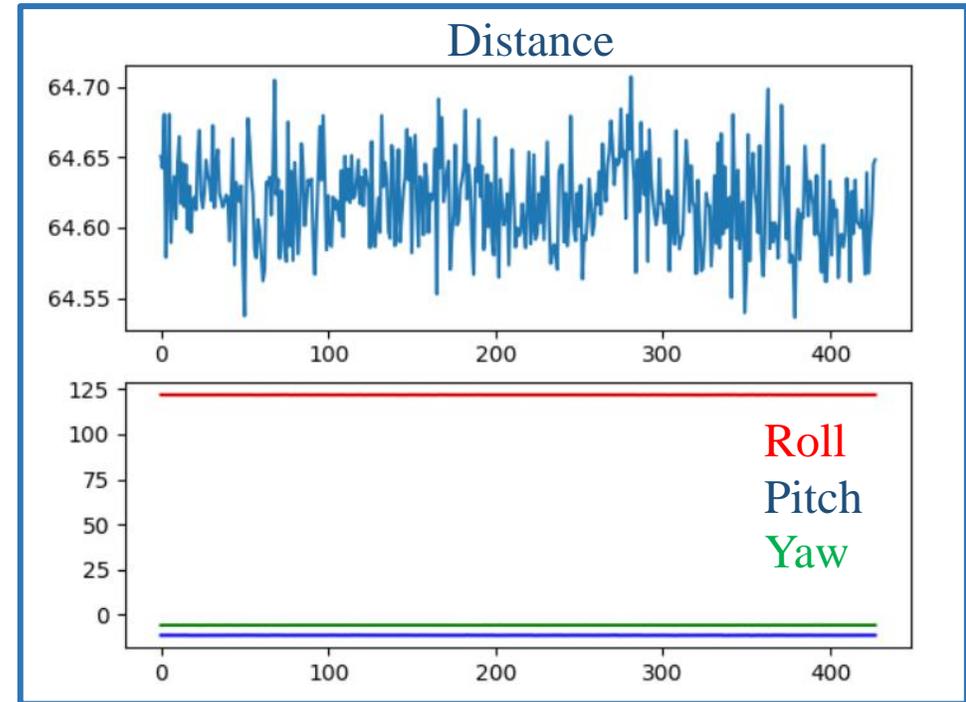
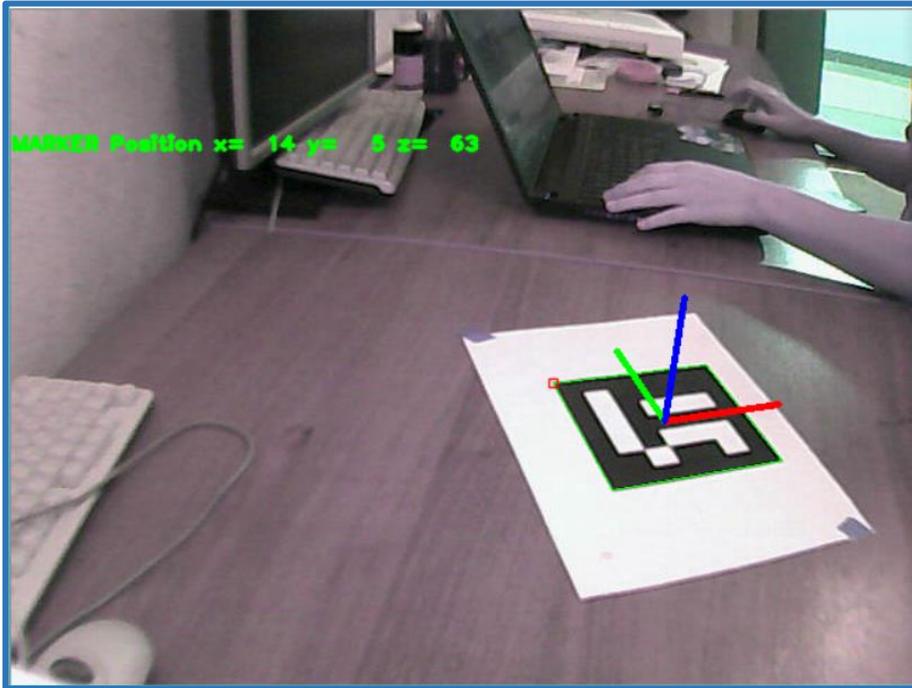
Систем координат изображения



Использование Aruco Pattern помогает установить соответствие между точкой из WCS и системой координат изображения



# Эксперимент



Расстояние :  $64.62\text{cm}$  стандартное отклонение = 0.03

Вращение ( $deg$ ) : 121.76 стандартное отклонение = 0.04

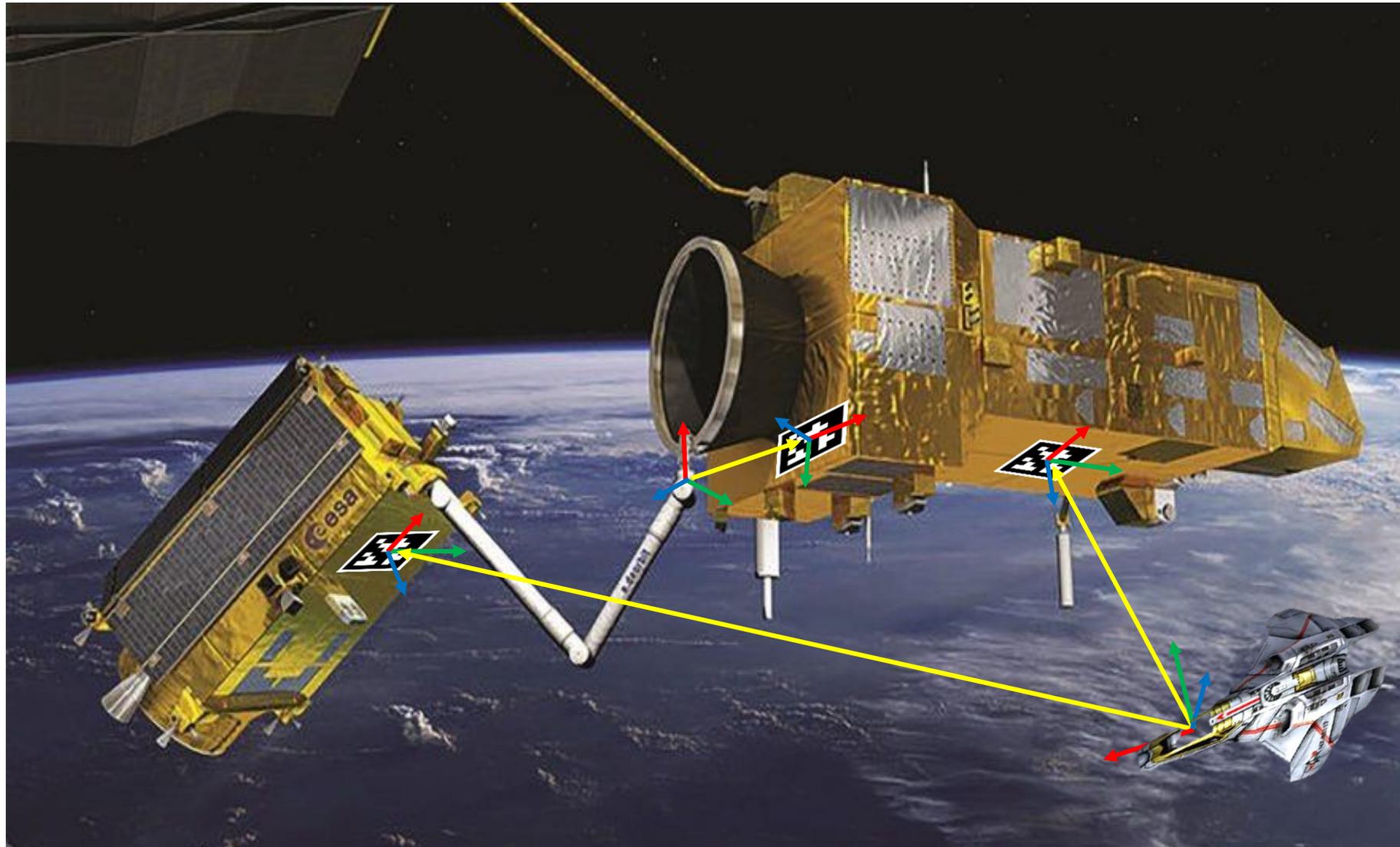
Крен ( $deg$ ) :  $-11.73$  стандартное отклонение = 0.034

Курс ( $deg$ ) :  $-5.759$  стандартное отклонение = 0.03

## Заключения

- Были получены довольно точные результаты при использовании изображений 480x640 пикселей (меньше 0.5%). При использовании камеры более высокого разрешения ожидаются более точные результаты.
- Необходимо в дальнейшем использовать методы очистки от шума, поскольку он был обнаружен при измерении данных.
- Вероятность ошибки в матрице перехода повышается при неправильном определении соответствия точек.

## Возможные применения на маленькие расстояния

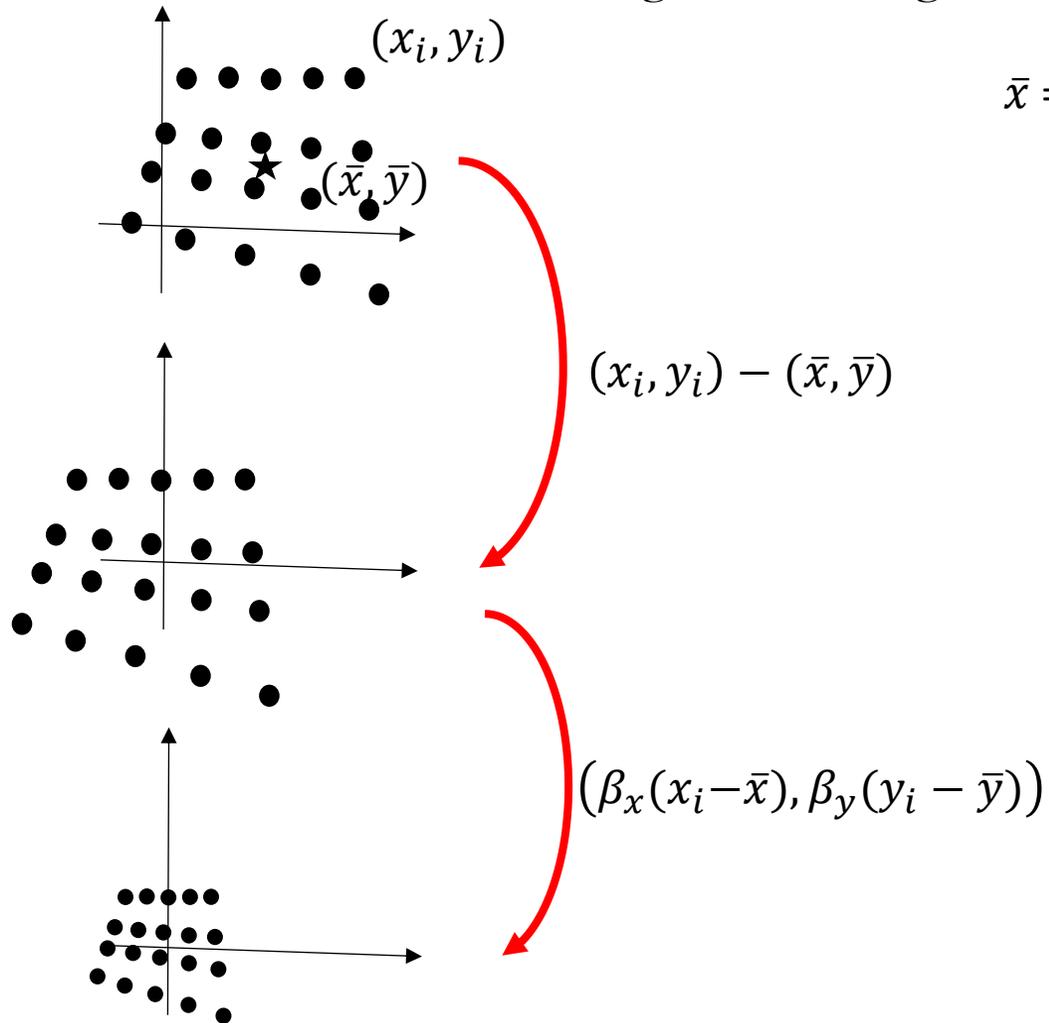


Background figure Source : <https://spacenews.com>

Спасибо за внимание

# Preconditioning

Вычисление преобразование  $T$  (перенос и изменение масштаба), которое превращает множество точек во множество. Так, что центроид нового множества начало координат  $(0,0)$  и среднее расстояние от начала координат  $\sqrt{2}$ . ( подробности “Multiview geometry in Computer Vision, page 109” and “In Defense of the Eight-Point Algorithm” Richard I. Hartley)



$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \quad \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i$$

$$\beta_x = \frac{1}{\frac{1}{N} \sum_{i=1}^N |x_i - \bar{x}|}$$

$$\beta_y = \frac{1}{\frac{1}{N} \sum_{i=1}^N |y_i - \bar{y}|}$$

$$\beta_x = \sqrt{\frac{1}{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}}$$

$$\beta_y = \sqrt{\frac{1}{\frac{1}{N} \sum_{i=1}^N (y_i - \bar{y})^2}}$$

$$H_{prec} = \begin{bmatrix} \beta_x & 0 & -\beta_x \bar{x} \\ 0 & \beta_y & -\beta_y \bar{y} \\ 0 & 0 & 1 \end{bmatrix}$$

Данная матрица  
вычисляется для каждого  
изображения