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
к.ф.-м.н., доц. С.С. Ткачев

г. Москва
2020

Содержание

Введение I

Было проведено множество исследований по измерению параметров движения объектов с помощью камер, где большое значение имело развитие методов калибровки камеры. Два исследования в области калибровки камеры примечательны тем, что, помимо определения внутренних параметров камеры, в процессе калибровки определяется положение трехмерного объекта в пространстве:

- Tsai[1] представил метод калибровки трехмерной (3D) камеры для метрологии машинного зрения, основанный на преобразовании трехмерных координат в изображение. где матрица поборота параметризована с использованием углов Эйлера.

$$R = \begin{bmatrix} \cos\psi\cos\theta & \sin\psi\cos\theta & -\sin\theta \\ -\sin\psi\cos\phi + \cos\psi\sin\theta\cos\phi & \cos\psi\cos\phi + \sin\psi\sin\theta\sin\phi & \cos\theta\sin\phi \\ \sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi & -\cos\psi\sin\phi + \sin\psi\sin\theta\cos\phi & \cos\theta\cos\phi \end{bmatrix}$$

- Zhang[2] представил метод, который также основан на преобразовании трехмерных координат в изображение, где матрица поборота параметризуется с использованием Формулы поворота Родрига.

$$R(\alpha, \boldsymbol{u}) = I_3\cos\alpha + (1 - \cos\alpha)\boldsymbol{u}\boldsymbol{u}^T + [\boldsymbol{u}]_x \sin\alpha$$

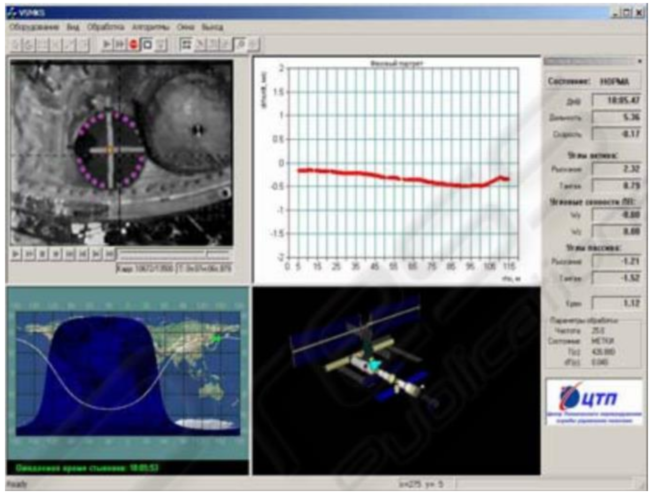
[1] R. Y. Tsai, “*A Versatile Camera Calibration Techniaue for High-Accuracy 3D Machine Vision Metrology Using Off-the-shelf TV Cameras and Lenses,* IEEE Journal on Robotics and Automation, 1987, Vol. 3, No. 4, pp. 323 – 344.

[2] Z. Zhang. *Flexible Camera Calibration By Viewing a Plane From Unknown Orientations*, Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, Greece, September 20-27,1999.

Введение II

Несколько лет назад возрос интерес к оценке параметров движения при обработке изображений для космических приложений:

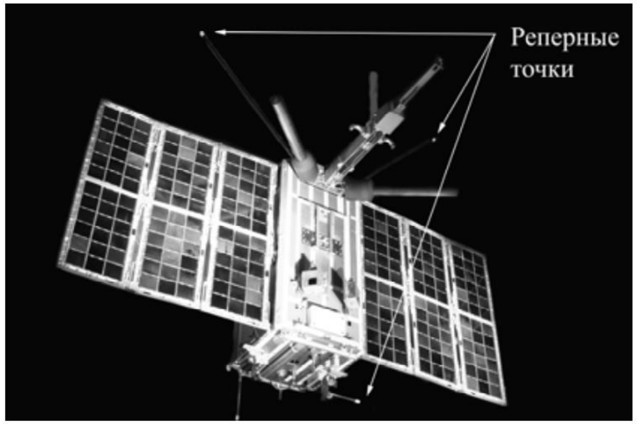
A.A.Boguslavsky[3] представил программный пакет, который с помощью видеосигнала, полученного от телекамеры, установленной на борту космического корабля, позволяет осуществлять автоматический визуальный мониторинг стыковки корабля "Прогресс" с Международной космической станцией.



При реальной стыковке матрица перехода представляется следующим образом

$$\|a_{ij}\| = \begin{Vmatrix} 1 & -\varphi_3 & \varphi_2 \\ -\varphi_3 & -1 & \varphi_1 \\ \varphi_2 & -\varphi_1 & -1 \end{Vmatrix}$$

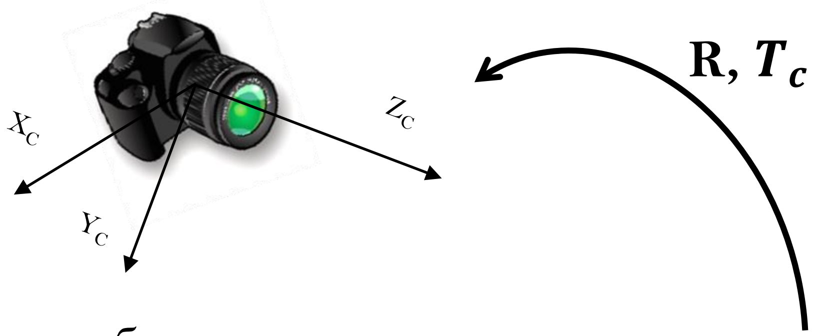
D. Ivanov[4], предложил алгоритм определения относительного положения и ориентации спутника путем обработки изображений космического аппарата, освещенного солнцем. Этот алгоритм использовался для определения относительного движения микроспутника ChibisM.



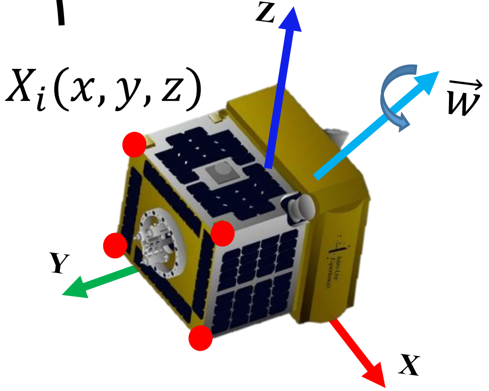
[3] A.A.Boguslavsky, V.V.Sazonov, S.M.Sokolov, A.I.Smirnov, Kh.S.Saigiraev. *Automatic vision-based monitoring of the spacecraft docking approach with the International Space Station*, Proceedings of the First International Conference on Informatics in Control, Automation and Robotics, 2004, pp. 79-86

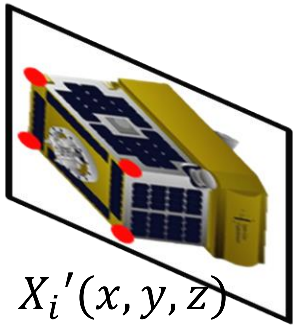
[4] Иванов Д. С., Карпенко С.О., Овчинников М. Ю., Сакович М.А. *Определение относительного движения спутников при их разделении по результатам обработки видеоизображения*, Препринт ИПМ им. М.В. Келдыша, 2012, № 57, 24 с.

Постановка задачи



доступное изображение, в котором можно определить положение точек $X_i(x, y, z)$ на изображении.





$X_i(x, y, z)$:местоположение конкретных точек объекта относительно его связанной системы координат XYZ известно

Необходимо определить ориентацию и угловой скорости объекта одновременно, используя следующие данные:

X_i: Точки объекта относительно связанной системы координат XYZ

X_i': Точки X_i, в проекции на оси системы координат изображения

Для этого необходимо:

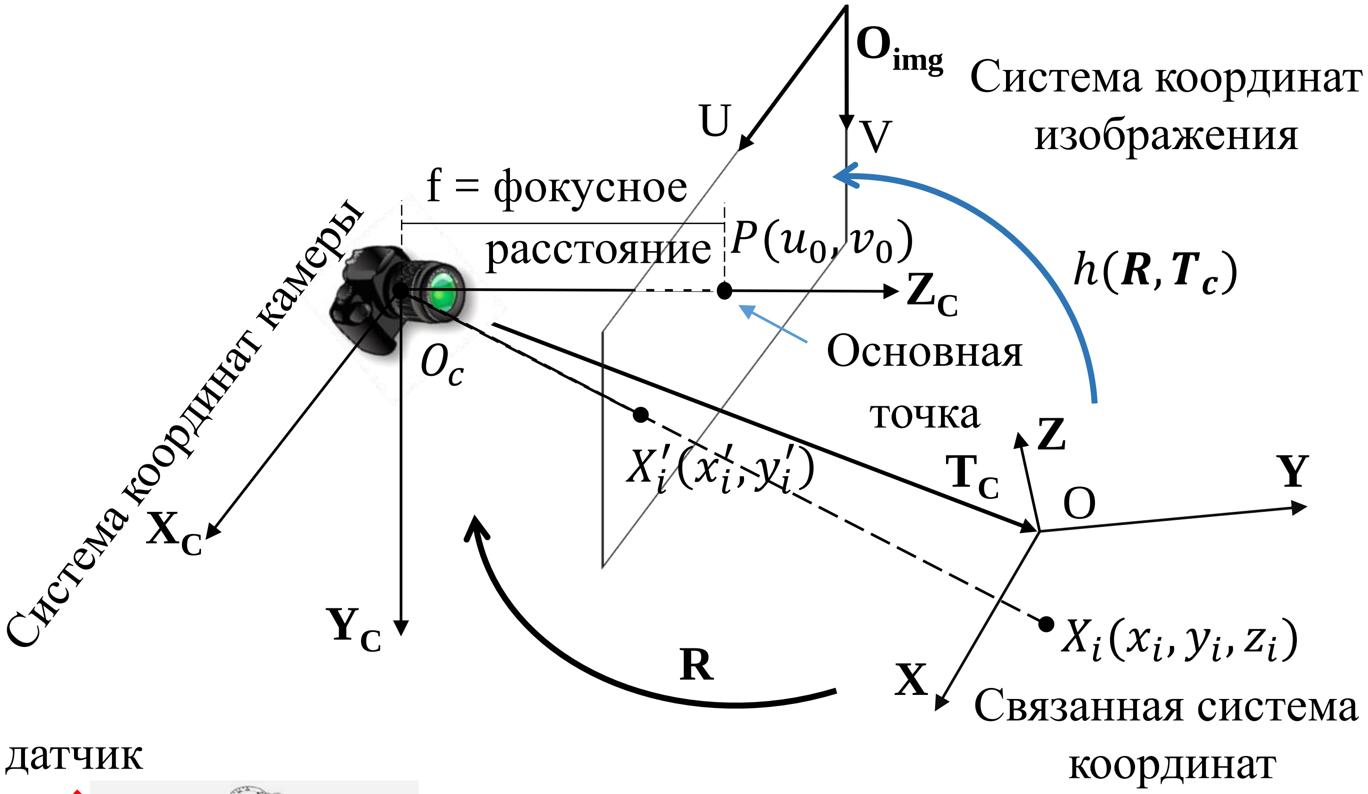
- Найти некоторую функцию, которая выполняет проекцию точек в систему координат изображения[1,2] и адаптировать его для использования кватернионов
- Определить угловую скорость с помощью

[1] R. Y. Tsai, "*A Versatile Camera Calibration Techniaue for High-Accuracy 3D Machine Vision Metrology Using Off-the-shelf TV Cameras and Lenses*, IEEE Journal on Robotics and Automation, 1987, Vol. 3, No. 4, pp. 323 – 344.

[2] Z. Zhang. *Flexible Camera Calibration By Viewing a Plane From Unknown Orientations*, Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, Greece, September 20-27,1999.

Элементы модели измерения

Модель камеры с точечным отверстием описывает



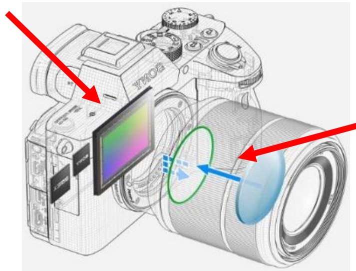
Описывает математическое соотношение между координатами точки в трехмерном пространстве и ее проекцией на плоскость изображения

Внутренние параметры (зависят только от свойств камеры):

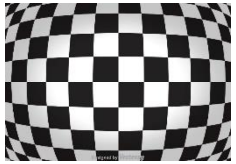
- f – фокусное расстояние (mm)
- (u_0, v_0) – Основная точка (pixel)
- α_x и α_y –размеры пикселя (pixel/mm)
- S – коэффициент скоса
- k_1, k_2, k_3, p_1, p_2 – коэффициенты искажения

Внешние параметры R и T_c

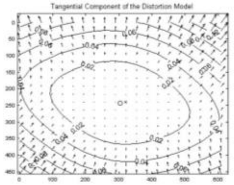
датчик



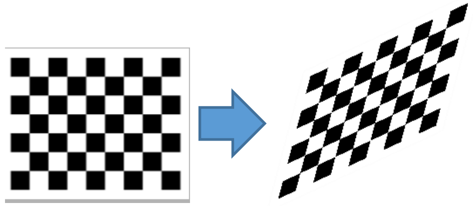
объективы



Радиальное Искажение



Тангенциальное Искажение



Скос
(Skew)

Модель измерения

Координаты точек переводятся из связанной системы координат в систему координат камеры

$$X_{c_i} = \begin{bmatrix} x_{c_i} \\ y_{c_i} \\ z_{c_i} \end{bmatrix} = R \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} + T_c$$

Проекция точки на плоскость датчика изображения

$$X_{cp_i} = \begin{bmatrix} x_{cp_i} \\ y_{cp_i} \end{bmatrix} = \begin{bmatrix} x_{c_i}/z_{c_i} \\ y_{c_i}/z_{c_i} \end{bmatrix}$$

Искажение (радиальное и тангенциальное) из-за объектива камеры[5]

$${r_i}^2 = {x_{cp_i}}^2 + {y_{cp_i}}^2$$

Координаты точек переводятся из плоскости датчика изображения в систему координат изображения

$$X_{p_i} = \begin{bmatrix} x_{p_i} \\ y_{p_i} \end{bmatrix} = \begin{bmatrix} (x_{d_i} + S y_{d_i}) \boxed{\alpha_x f} + u_0 \\ \boxed{\alpha_y f} y_{d_i} + v_0 \end{bmatrix} = \begin{bmatrix} (x_{d_i} + s y_{d_i}) f_x + u_0 \\ y_{d_i} f_y + v_0 \end{bmatrix}$$

Где α_x и α_y Ширина и высота пикселя

$$X_{p_i} = h\,(X_i, \boxed{f_x, f_y, u_0, v_0, S, k_1, k_2, k_3, p_1, p_2}, R, T_c)$$

h является функцией, которая выполняет проекцию точек в систему координат изображения

X_{p_i}: Проецируемая точка в системе координат изображения

h используемая для калибровки камеры, Главная цель калибровки – определение внутренних параметров, которые зависят только от свойств камеры.

После калибровки h зависит только от R и T_c

$$X_{d_i} = \begin{bmatrix} x_{d_i} \\ y_{d_i} \end{bmatrix} = \begin{bmatrix} x_{cp_i}(1 + k_1 {r_i}^2 + k_2 {r_i}^4 + k_3 {r_i}^6) + 2p_1 x_{cp_i} y_{cp_i} + p_2({r_i}^2 + 2{x_{cp_i}}^2) \\ y_{cp_i}(1 + k_1 {r_i}^2 + k_2 {r_i}^4 + k_3 {r_i}^6) + 2p_2 x_{cp_i} y_{cp_i} + p_1({r_i}^2 + 2{y_{cp_i}}^2) \end{bmatrix}$$

[5] D. C. Brown, *Decentering Distortion of Lenses*, *Photometric Engineering* 32(3), 1966, pp. 444–462.

Градиент модели измерения

В связи с тем, что модель измерения является составной функцией нескольких преобразований, включающих вектор и матрицы, удобно определять градиент с помощью матричного исчисления.

Пусть:

Фокусное расстояние : $F = [f_x \quad f_y]$(pixel)
Основная точка $C = [u_0 \quad v_0]$ (pixel)
Коэффициенты искажения : $K_D{=}[k_1 \quad k_2 \quad p_1 \quad p_2 \quad k_3]$
Коэффициент скоса : S

Модель измерения будет переписана следующим образом

$$X_{p_i} = h\ (X_i, f_x, f_y, u_0, v_0, S, k_1, k_2, k_3, p_1, p_2, R, T_c)$$

$$X_{p_i} = \begin{bmatrix} x_{p_i} \\ y_{p_i} \end{bmatrix} = h(X_i, F, C, S, K_D, \vec{q}, T_c)$$

Градиент модели измерения (для одной точки):

$$Dh = \begin{bmatrix} \frac{\partial h}{\partial F} & \frac{\partial h}{\partial C} & \frac{\partial h}{\partial S} & \frac{\partial h}{\partial K_D} & \frac{\partial h}{\partial \vec{q}} & \frac{\partial h}{\partial T_c} \end{bmatrix}$$

Градиент модели измерения

Производная по фокусному расстоянию

$$\frac{\partial h}{\partial F} = \begin{bmatrix} \frac{\partial x_{p_i}}{\partial F} \\ \frac{\partial y_{p_i}}{\partial F} \end{bmatrix} = \begin{bmatrix} x_{d_i} + s y_{d_i} & 0 \\ 0 & y_{d_i} \end{bmatrix}$$

Производная по координатам центральной точки изображения

$$\frac{\partial h}{\partial C} = \begin{bmatrix} \frac{\partial x_{p_i}}{\partial C} \\ \frac{\partial y_{p_i}}{\partial C} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Производная по коэффициенту скоса

$$\frac{\partial h}{\partial s} = \begin{bmatrix} \frac{\partial x_{p_i}}{\partial s} \\ \frac{\partial y_{p_i}}{\partial s} \end{bmatrix} = \begin{bmatrix} y_{p_i} f_x \\ 0 \end{bmatrix}$$

Производная по коэффициентам искажения изображения

$$\frac{\partial h}{\partial K_D} = \begin{bmatrix} \frac{\partial x_{p_i}}{\partial K_D} \\ \frac{\partial y_{p_i}}{\partial K_D} \end{bmatrix} = \begin{bmatrix} x_{cp_i}({x_{cp_i}}^2 + {y_{cp_i}}^2) & x_{cp_i}\left({x_{cp_i}}^2 + {y_{cp_i}}^2\right)^2 & 2x_{d_i} y_{d_i} & {y_{cp_i}}^2 + 3{x_{cp_i}}^2 & x_{cp_i}\left({x_{cp_i}}^2 + {y_{cp_i}}^2\right)^3 \\ y_{cp_i}({x_{cp_i}}^2 + {y_{cp_i}}^2) & y_{cp_i}\left({x_{cp_i}}^2 + {y_{cp_i}}^2\right)^2 & {x_{cp_i}}^2 + 3{y_{cp_i}}^2 & 2x_{d_i} y_{d_i} & y_{cp_i}\left({x_{cp_i}}^2 + {y_{cp_i}}^2\right)^3 \end{bmatrix}$$

Градиент модели измерения

$$\frac{\partial h}{\partial T_c} = \begin{bmatrix} \frac{\partial x_{p_i}}{\partial T_c} \\ \frac{\partial y_{p_i}}{\partial T_c} \end{bmatrix}$$

Производная по вектору переноса
(расстояния между камерой и объектом)

$$\frac{\partial x_{p_i}}{\partial T_c} = f_x\left(D_{r_i}\frac{\partial x_{cp_i}}{\partial T_c} + x_{cp_i}\left[k1\frac{\partial(r_i{}^2)}{\partial T_c} + k2\frac{\partial(r_i{}^4)}{\partial T_c} + k3\frac{\partial(r_i{}^6)}{\partial T_c}\right] + \left(2p_1 y_{cp_i} + 6p_2 x_{cp_i}\right)\frac{\partial x_{cp_i}}{\partial T_c} + \left(2p_1 x_{cp_i} + 2p_2 y_{cp_i}\right)\frac{\partial y_{cp_i}}{\partial T_c}\right)$$
$$+ sf_x\left(D_{r_i}\frac{\partial y_{cp_i}}{\partial T_c} + y_{cp_i}\left[k1\frac{\partial(r_i{}^2)}{\partial T_c} + k2\frac{\partial(r_i{}^4)}{\partial T_c} + k3\frac{\partial(r_i{}^6)}{\partial T_c}\right] + \left(2p_2 y_{cp_i} + 2p_1 x_{cp_i}\right)\frac{\partial x_{cp_i}}{\partial T_c} + \left(2p_2 x_{cp_i} + 6p_1 y_{cp_i}\right)\frac{\partial y_{cp_i}}{\partial T_c}\right)$$

$$\frac{\partial y_{p_i}}{\partial T_c} = f_y\left(D_{r_i}\frac{\partial y_{cp_i}}{\partial T_c} + y_{cp_i}\left[k1\frac{\partial(r_i{}^2)}{\partial T_c} + k2\frac{\partial(r_i{}^4)}{\partial T_c} + k3\frac{\partial(r_i{}^6)}{\partial T_c}\right] + \left(2p_2 y_{cp_i} + 2p_1 x_{cp_i}\right)\frac{\partial x_{cp_i}}{\partial T_c} + \left(2p_2 x_{cp_i} + 6p_1 y_{cp_i}\right)\frac{\partial y_{cp_i}}{\partial T_c}\right)$$

$$\frac{\partial(r_i{}^2)}{\partial T_c} = 2\begin{bmatrix} x_{cp_i} & y_{cp_i} \end{bmatrix}\frac{\partial X_{cp_i}}{\partial T_c}$$

$$\frac{\partial(r_i{}^4)}{\partial T_c} = 2r_i{}^2\frac{\partial(r_i{}^2)}{\partial T_c} = 4\left(x_{cp_i}{}^2 + y_{cp_i}{}^2\right)\begin{bmatrix} x_{cp_i} & y_{cp_i} \end{bmatrix}\frac{\partial X_{cp_i}}{\partial T_c}$$

$$\frac{\partial(r_i{}^6)}{\partial T_c} = 3r_i{}^4\frac{\partial(r_i{}^2)}{\partial T_c} = 6\left(x_{cp_i}{}^2 + y_{cp_i}{}^2\right)^2\begin{bmatrix} x_{cp_i} & y_{cp_i} \end{bmatrix}\frac{\partial X_{cp_i}}{\partial T_c}$$

$$\frac{\partial X_{cp_i}}{\partial T_c} = \begin{bmatrix} \frac{\partial x_{cp_i}}{\partial T_c} \\ \frac{\partial y_{cp_i}}{\partial T_c} \end{bmatrix} = \begin{bmatrix} \frac{1}{z_{c_i}} & 0 & -\frac{x_{c_i}}{z_{c_i}{}^2} \\ 0 & \frac{1}{z_{c_i}} & -\frac{y_{c_i}}{z_{c_i}{}^2} \end{bmatrix}$$

Градиент модели измерения

$$\frac{\partial h}{\partial \vec{q}} = \frac{\partial h}{\partial R}\frac{\partial R(\vec{q})}{\partial \vec{q}}$$

Производная по ориентации объекта
(ориентация выражается через кватернионы)

$$\frac{\partial h}{\partial R} = \begin{bmatrix} \frac{\partial x_{p_i}}{\partial R} \\ \frac{\partial y_{p_i}}{\partial R} \end{bmatrix}$$

$$\frac{\partial x_{p_i}}{\partial R} = f_x\left(D_{r_i}\frac{\partial x_{cp_i}}{\partial R} + x_{cp_i}\left[k1\frac{\partial({r_i}^2)}{\partial R} + k2\frac{\partial({r_i}^4)}{\partial R} + k3\frac{\partial({r_i}^6)}{\partial R}\right] + \left(2p_1 y_{cp_i} + 6p_2 x_{cp_i}\right)\frac{\partial x_{cp_i}}{\partial R} + \left(2p_1 x_{cp_i} + 2p_2 y_{cp_i}\right)\frac{\partial y_{cp_i}}{\partial R}\right) + sf_x\left(D_{r_i}\frac{\partial y_{cp_i}}{\partial R} + y_{cp_i}\left[k1\frac{\partial({r_i}^2)}{\partial R} + k2\frac{\partial({r_i}^4)}{\partial R} + k3\frac{\partial({r_i}^6)}{\partial R}\right] + \left(2p_2 y_{cp_i} + 2p_1 x_{cp_i}\right)\frac{\partial x_{cp_i}}{\partial R} + \left(2p_2 x_{cp_i} + 6p_1 y_{cp_i}\right)\frac{\partial y_{cp_i}}{\partial R}\right)$$

$$\frac{\partial y_{p_i}}{\partial R} = f_y\left(D_{r_i}\frac{\partial y_{cp_i}}{\partial R} + y_{cp_i}\left[k1\frac{\partial({r_i}^2)}{\partial R} + k2\frac{\partial({r_i}^4)}{\partial R} + k3\frac{\partial({r_i}^6)}{\partial R}\right] + \left(2p_2 y_{cp_i} + 2p_1 x_{cp_i}\right)\frac{\partial x_{cp_i}}{\partial R} + \left(2p_2 x_{cp_i} + 6p_1 y_{cp_i}\right)\frac{\partial y_{cp_i}}{\partial R}\right)$$

$$D_{r_i} = 1 + k_1 {r_i}^2 + k_2 {r_i}^4 + k_3 {r_i}^6$$

Градиент модели измерения

$$\frac{\partial(r_i{}^2)}{\partial R} = 2\begin{bmatrix} x_{cp_i} & y_{cp_i}\end{bmatrix}\frac{\partial X_{cp_i}}{\partial R}$$

$$\frac{\partial(r_i{}^4)}{\partial R} = 2r_i{}^2\frac{\partial(r_i{}^2)}{\partial R} = 4\left(x_{cp_i}{}^2 + y_{cp_i}{}^2\right)\begin{bmatrix} x_{cp_i} & y_{cp_i}\end{bmatrix}\frac{\partial X_{cp_i}}{\partial R}$$

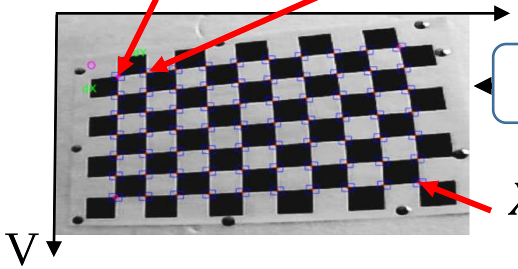
$$\frac{\partial(r_i{}^6)}{\partial R} = 3r_i{}^4\frac{\partial(r_i{}^2)}{\partial R} = 6\left(x_{cp_i}{}^2 + y_{cp_i}{}^2\right)^2\begin{bmatrix} x_{cp_i} & y_{cp_i}\end{bmatrix}\frac{\partial X_{cp_i}}{\partial R}$$

$$\frac{\partial R}{\partial \vec{q}} = \begin{bmatrix} 0 & -4\lambda_2 & -4\lambda_3 \\ 2\lambda_2 - 2\lambda_3\,\lambda_1/\lambda_0 & 2\lambda_1 - 2\lambda_3\,\lambda_2/\lambda_0 & 2\lambda_0 - 2\lambda_3\,\lambda_3/\lambda_0 \\ 2\lambda_3 + 2\lambda_2\,\lambda_1/\lambda_0 & -2\lambda_0 + \lambda_2\,\lambda_2/\lambda_0 & 2\lambda_1 + 2\lambda_2\,\lambda_3/\lambda_0 \\ 2\lambda_2 + 2\lambda_3\,\lambda_1/\lambda_0 & 2\lambda_1 + 2\lambda_3\,\lambda_2/\lambda_0 & -2\lambda_0 + 2\lambda_3\,\lambda_3/\lambda_0 \\ -4\lambda_1 & 0 & -4\lambda_3 \\ 2\lambda_0 - 2\lambda_1\,\lambda_1/\lambda_0 & 2\lambda_3 - 2\lambda_1\,\lambda_2/\lambda_0 & 2\lambda_2 - 2\lambda_1\,\lambda_3/\lambda_0 \\ 2\lambda_3 - 2\lambda_2\,\lambda_1/\lambda_0 & 2\lambda_0 - 2\lambda_2\,\lambda_2/\lambda_0 & 2\lambda_1 - 2\lambda_2\,\lambda_3/\lambda_0 \\ -2\lambda_0 + 2\lambda_1\,\lambda_1/\lambda_0 & 2\lambda_3 + 2\lambda_1\,\lambda_2/\lambda_0 & 2\lambda_2 + 2\lambda_1\,\lambda_3/\lambda_0 \\ -4\lambda_1 & -4\lambda_2 & 0 \end{bmatrix}$$

Алгоритм для калибровки

Изображения 1

$X'^1_1 = (x'^1_1, y'^1_1)$ $X'^1_2 = (x'^1_2, y'^1_2)$

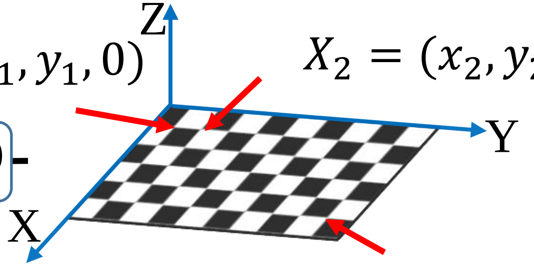


$h^1(F, C, S, K_D, \vec{q}_1, T_{c_1})$

$X'^1_n = (x'^1_n, y'^1_n)$

связанная система координат

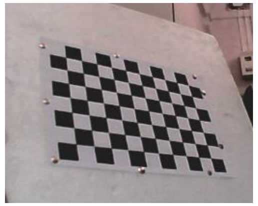
$X_1 = (x_1, y_1, 0)$ $X_2 = (x_2, y_2, 0)$



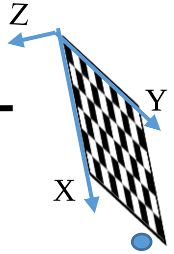
$X_n = (x_n, y_n, 0)$

аналогично

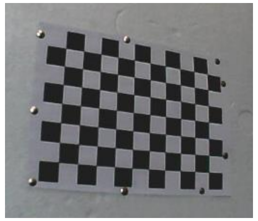
Изображения 2



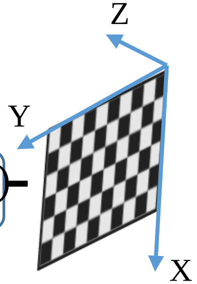
$h^2(F, C, S, K_D, \vec{q}_2, T_{c_2})$



Изображения m



$h^m(F, C, S, K_D, \vec{q}_2, T_{c_2})$



$k = 1,2, \ldots, m$

h^k: модель измерения, котороя проекцирует точки X_i на k-е изображение

X'^k_i :Точка X_i в k-ом изображении

вектор ошибок :

$$\epsilon = \begin{bmatrix} X'^1_1 - h(X_1, F, C, S, K_D, \vec{q}_1, T_{c_1}) \\ X'^1_2 - h(X_2, F, C, S, K_D, \vec{q}_1, T_{c_1}) \\ \vdots \\ X'^1_n - h(X_n, F, C, S, K_D, \vec{q}_1, T_{c_1}) \\ X'^2_1 - h(X_1, F, C, S, K_D, \vec{q}_2, T_{c_2}) \\ X'^2_2 - h(X_2, F, C, S, K_D, \vec{q}_2, T_{c_2}) \\ \vdots \\ X'^2_n - h(X_n, F, C, S, K_D, \vec{q}_2, T_{c_2}) \\ \vdots \\ X'^k_i - h(X_i, F, C, S, K_D, \vec{q}_k, T_{c_k}) \\ \vdots \\ X'^m_1 - h(X_1, F, C, S, K_D, \vec{q}_m, T_{c_m}) \\ X'^m_2 - h(X_2, F, C, S, K_D, \vec{q}_m, T_{c_m}) \\ \vdots \\ X'^m_n - h(X_n, F, C, S, K_D, \vec{q}_m, T_{c_m}) \end{bmatrix}$$

Алгоритм для калибровки

Пусть h^k:

$$h^k = \begin{bmatrix} h(X_1, F, C, S, K_D, \vec{q}_k, T_{c_k}) \\ h(X_2, F, C, S, K_D, \vec{q}_k, T_{c_k}) \\ \vdots \\ h(X_n, F, C, S, K_D, \vec{q}_k, T_{c_k}) \end{bmatrix}$$

Пусть X'^k – векторное представлением всех точек в k-ом изображении

$$X'^k = \left[x'^k_1, y'^k_1, x'^k_2, y'^k_2, \dots, x'^k_n, y'^k_n\right]^T$$

Вектор ошибок будет переписана следующим образом

$$\epsilon(M) = \begin{bmatrix} X'^1 - h^1(F, C, S, K_D, \vec{q}_1, T_{c_1}) \\ X'^2 - h^2(F, C, S, K_D, \vec{q}_2, T_{c_2}) \\ \vdots \\ X'^{\mathrm{m}} - h^m(F, C, S, K_D, \vec{q}_m, T_{c_m}) \end{bmatrix}$$

Параметров каливровки: $M = \left[F, C, S, K_D, \vec{q}_1, T_{c_1}, \dots, \vec{q}_k, T_{c_k}\right]$

Задача : $\min|\epsilon(M)|^2$

Алгоритм Гаусса - Ньютона :

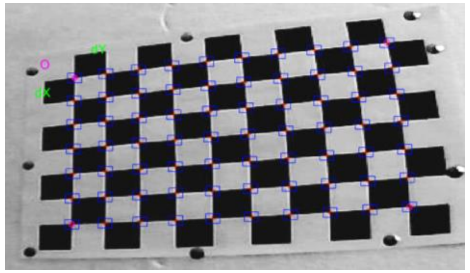
$\epsilon_s = \epsilon(M_s)$ $\qquad M_0, \epsilon_0 = \epsilon(M_0)$

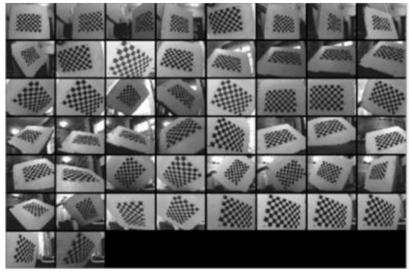
$M_{s+1} = M_s + \Delta M$, где $\Delta M = (J^T J)^{-1} J^T \epsilon_s$

Матрица Якоби

$$J = \frac{\partial \epsilon}{\partial M} = \begin{bmatrix} \frac{\partial h^1}{\partial M} \\ \frac{\partial h^2}{\partial M} \\ \vdots \\ \frac{\partial h^m}{\partial M} \end{bmatrix} = \begin{bmatrix} \frac{\partial h^1}{\partial F} & \frac{\partial h^1}{\partial C} & \frac{\partial h^1}{\partial S} & \frac{\partial h^1}{\partial K_D} & \frac{\partial h^1}{\partial \vec{q}_1} & \frac{\partial h^1}{\partial T_{c_1}} & 0_{2\mathrm{nx}3} & 0_{2\mathrm{nx}3} & \dots & 0_{2\mathrm{nx}3} & 0_{2\mathrm{nx}3} \\ \frac{\partial h^2}{\partial F} & \frac{\partial h^2}{\partial C} & \frac{\partial h^2}{\partial S} & \frac{\partial h^2}{\partial K_D} & 0_{2\mathrm{nx}3} & 0_{2\mathrm{nx}3} & \frac{\partial h^2}{\partial \vec{q}_2} & \frac{\partial h^2}{\partial T_{c_2}} & \dots & 0_{2\mathrm{nx}3} & 0_{2\mathrm{nx}3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & 0_{2\mathrm{nx}3} & 0_{2\mathrm{nx}3} \\ \frac{\partial h^m}{\partial F} & \frac{\partial h^m}{\partial C} & \frac{\partial h^m}{\partial S} & \frac{\partial h^m}{\partial K_D} & 0_{2\mathrm{nx}3} & 0_{2\mathrm{nx}3} & 0_{2\mathrm{nx}3} & 0_{2\mathrm{nx}3} & \dots & \frac{\partial h^m}{\partial \vec{q}_{\mathrm{m}}} & \frac{\partial h^m}{\partial T_{c_{\mathrm{m}}}} \end{bmatrix}$$

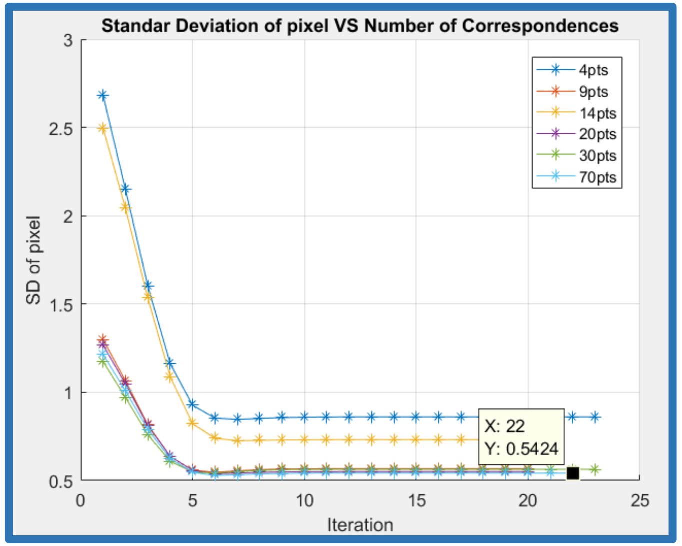
Результаты калибровки

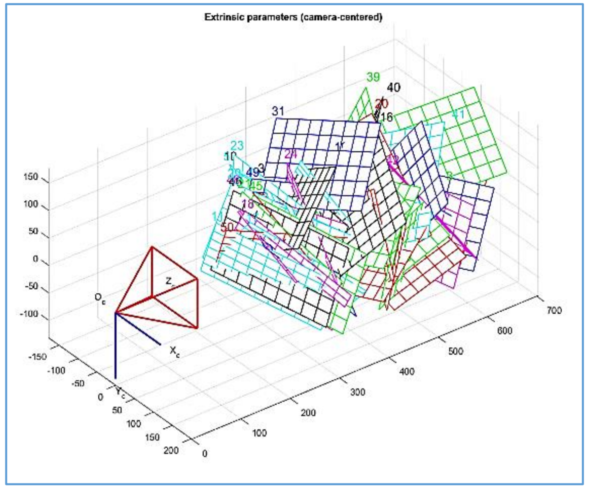




50 изображений для калибровки,70 точек для одного изображения

Количество параметров: 310
Матрица Якобы: 7000x310





Визуализация внешних параметров

Внутренние параметры: Model FI8918W, 480x640 pixels , 15fps

Фокусное расстояние:

$fx = 635.61266$ (pixel)

$fy = 636.45016$ (pixel)

Skew: $S = 0.0$

Основная точка

$uo = 320.88665$(pixel)

$vo = 246.79063$ (pixel)

искажение : $\boldsymbol{K_1} = -0.44492$, $\boldsymbol{K_2} = 0.25395$, $\boldsymbol{p_1} = 0.00246$, $\boldsymbol{p_2} = -0.00378$, $\boldsymbol{K_3} = 0.000$

SD : 0.54 pixel

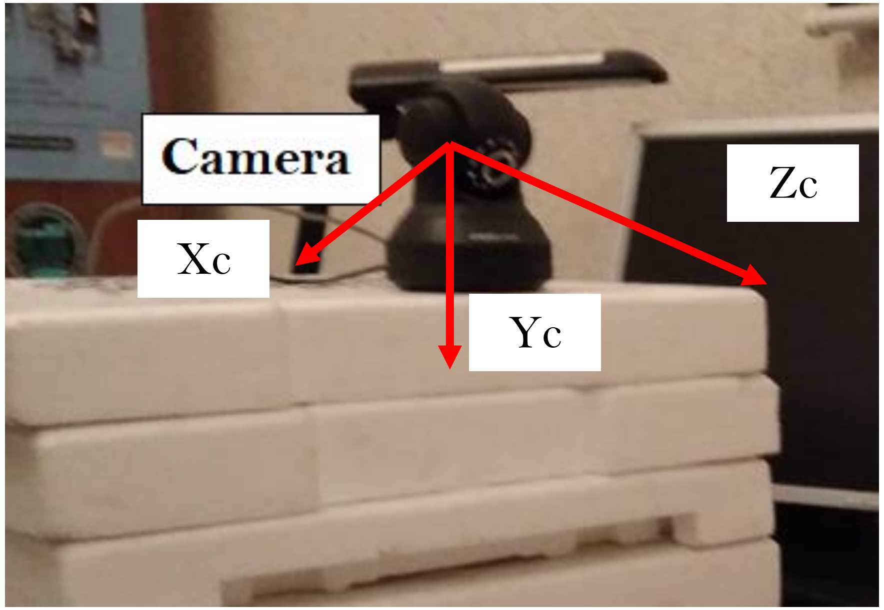
Калибровочное программное[6] использовалось потому, что оно дает возможность выбирать точки каждого изображения. Этот инструмент основан на формуле вращения Формулы поворота Родрига

Важно отметить, что хотя во время калибровки значения внутренних и внешних параметров оптимизируются. Основная цель состоит в том, чтобы определить внутренние параметры

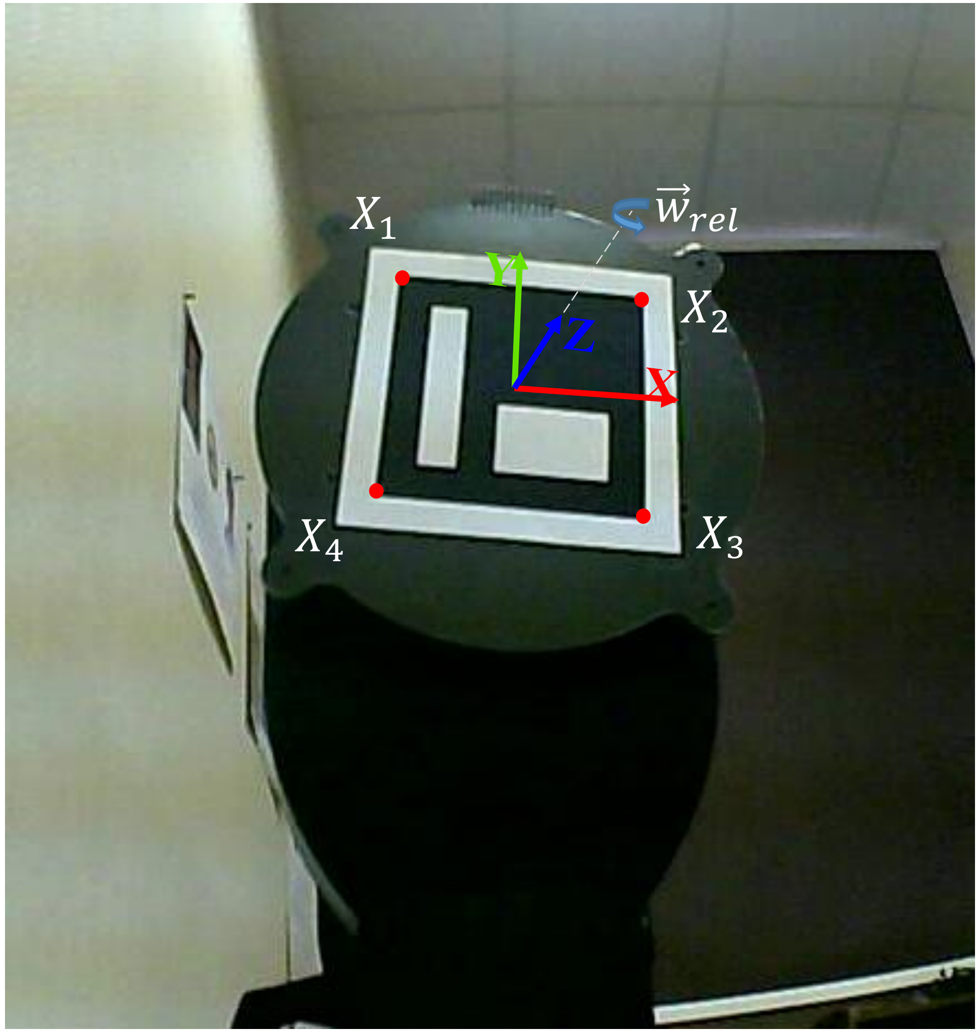
[6]Bouguet, J. Y. Camera Calibration Toolbox for Matlab, 2015, Available: http://www.vision.caltech.edu/bouguetj/calib_doc/ (дата обращения: 20.06.220).

Опыт : Поворот на 90 градусов

Использование Aruco marker[8] помогает установить соответствие между точкой из связанной системы координат и системой координат изображения



Необходимо знать размер Aruco marker

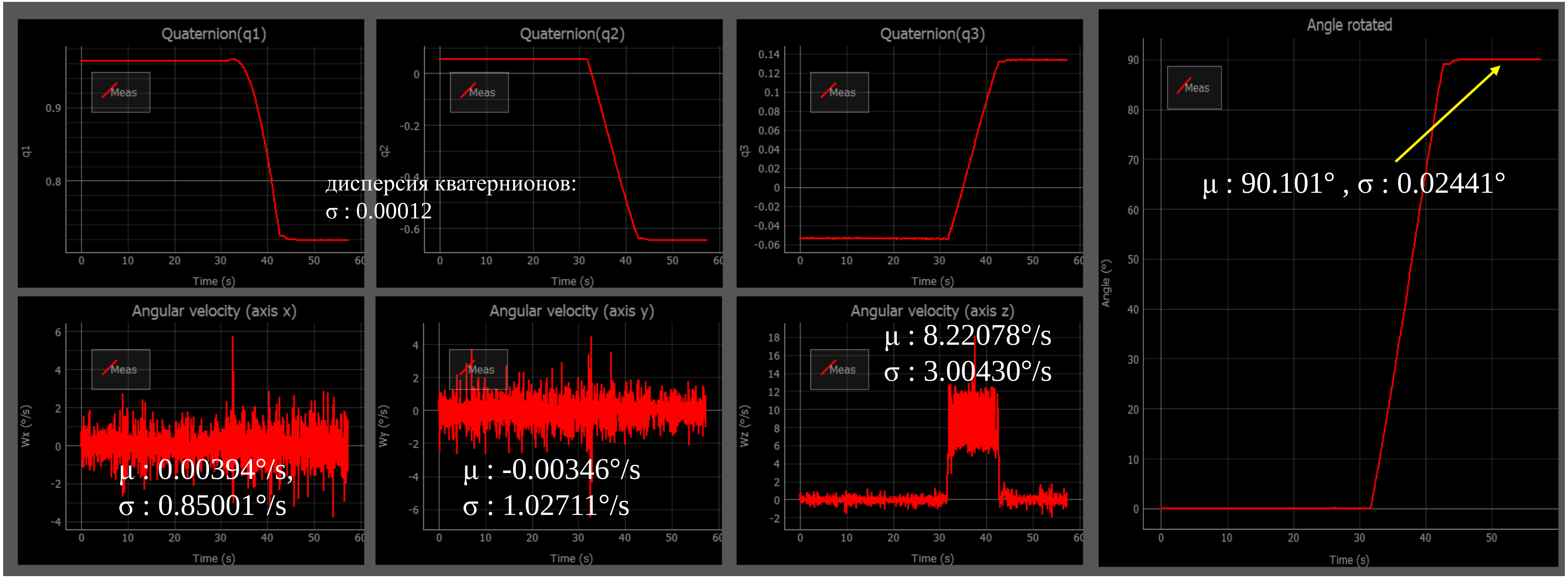


Поворотный стол вращается по оси Z

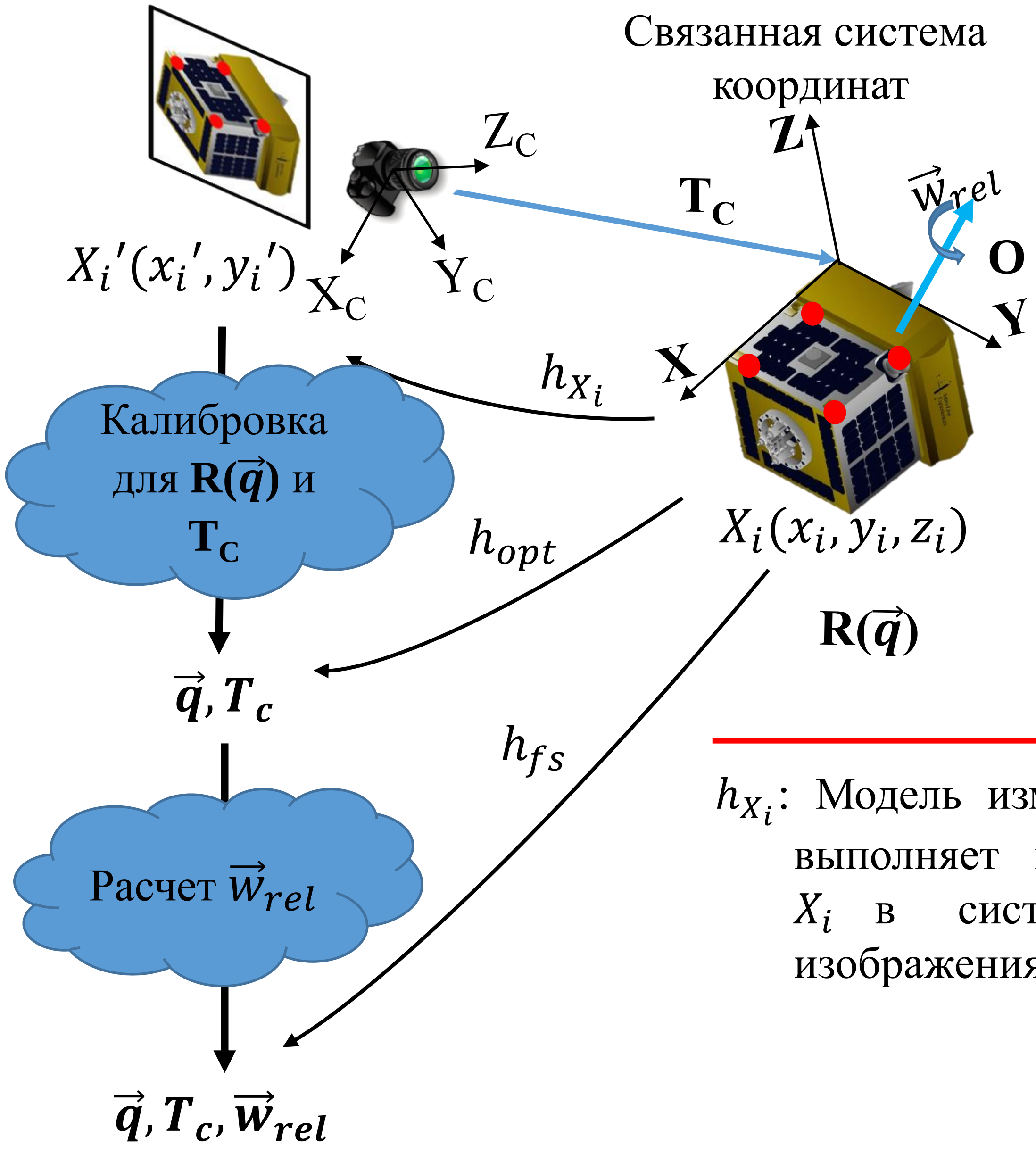
[7]F. J. Romero-Ramirez, R. Muñoz-Salinas, R. Medina-Carnicer. *Speeded up detection of squared fiducial markers*, Image and Vision Computing, 2018, vol 76, pages 38-47

Опыт : Поворот на 90 градусов

Измерение ориентации с использованием модели измерения и процесса калибровки только для ориентации



Пространство состояний системы



h_{opt}: Определяет $\vec{q}$ и T_c с помощью итерационного процесса, это требует больше вычислительного времени

Модель наблюдения для модели состояния пространства:

$$Z = \begin{bmatrix} \vec{q} \\ T_c \end{bmatrix} = h_{opt}(\vec{q}, T_c)$$

h_{fs}: Добавляет измерения угловой скорости, полученные из последовательных ориентации $\mathbf{R}(\vec{q})$ объекта, измеренных в каждый период времени Δt

Модель наблюдения для модели состояния пространства:

$$\begin{bmatrix} \vec{q} \\ T_c \\ \vec{w} \end{bmatrix} = h_{fs}(\vec{q}, T_c, \vec{w})$$

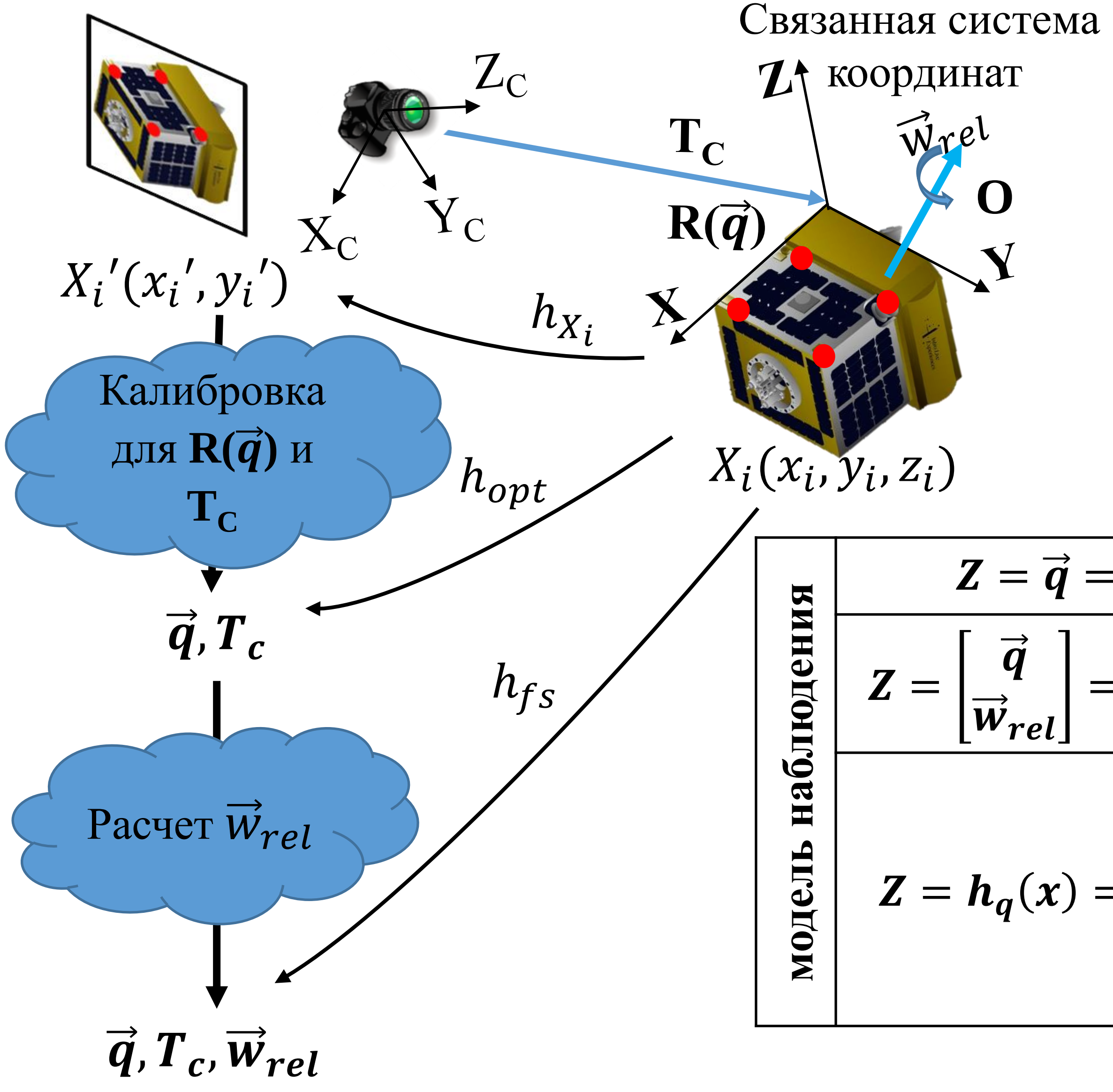
---

h_{X_i}: Модель измерения, которая выполняет проекцию точки X_i в систему координат изображения

Модель наблюдения для модели состояния пространства системы

$$Z = h_q(x) = \begin{bmatrix} h_{X_1}(\vec{q}, T_c) \\ h_{X_2}(\vec{q}, T_c) \\ h_{X_3}(\vec{q}, T_c) \\ h_{X_4}(\vec{q}, T_c) \end{bmatrix}$$

Пространство состояний системы



		модель процесса	
		Непрерывное время	**Дискретное время**
		$\dot{\Lambda} = \frac{1}{2}\Lambda \circ \vec{w}_{rel}$ $\dot{w}_{rel} = 0_{3x1}$	$\Lambda_k = [I_{4x4} + \frac{1}{2}\Psi_{k-1}\Delta t]\Lambda_{k-1}$ $\vec{w}_{rel_k} = \vec{w}_{rel_{k-1}}$
модель наблюдения	$\boldsymbol{Z = \vec{q} = h_{opt}(\vec{q})}$	не работает	ok
	$\boldsymbol{Z = \begin{bmatrix} \vec{q} \\ \vec{w}_{rel} \end{bmatrix} = h_{fs}(\vec{q}, \vec{w}_{rel})}$	ok	ok
	$\boldsymbol{Z = h_q(x)} = \begin{bmatrix} h_{X_1}(\vec{q}, T_c) \\ h_{X_2}(\vec{q}, T_c) \\ h_{X_3}(\vec{q}, T_c) \\ h_{X_4}(\vec{q}, T_c) \end{bmatrix}$	$\boldsymbol{h_q}$:Сильно нелинейно	$\boldsymbol{h_q}$:Сильно нелинейно

Задание матриц для работы непрерывного EKF

Вектор состояния $\boldsymbol{x} = \begin{bmatrix} \vec{q} \\ \vec{w} \end{bmatrix}$ $\quad \Lambda = [q_0, q_1, q_2, q_3]^T = \begin{bmatrix} q_0 \\ \vec{q} \end{bmatrix}$

$\vec{w} = [w_x, w_y, w_z]^T$

Модель движения $\dot{\Lambda} = \frac{1}{2} \Lambda \circ \vec{w}_{rel}, \qquad |\Lambda| = 1$

$\dot{w}_{rel} = 0_{3x1}$

Модель измерений $\boldsymbol{z_K} = \boldsymbol{H x_k} = \boldsymbol{h}_{fs}(\vec{\boldsymbol{q}}, \vec{\boldsymbol{w}}) \quad \boldsymbol{H} = \boldsymbol{I_6}$

Начальное значение матрицы ошибок P(0) =I_{6x6}

Линеаризация уравнений движения

$$\frac{d}{dt} \delta \boldsymbol{x}(t) = \boldsymbol{F}(t) \delta \boldsymbol{x}(t)$$

$$\boldsymbol{F} = \begin{bmatrix} -[\vec{w}_{rel}]_x & 0.5\boldsymbol{I}_{3\boldsymbol{x}3} \\ 0_{3\boldsymbol{x}3} & 0_{3\boldsymbol{x}3} \end{bmatrix}$$

$$[\vec{w}_{rel}]_x = \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{bmatrix}$$

Матрица щума измерений

$$R = diag([6.123e-4,\ 1.318e-8,\ \ 3.377e-7, 2.73e-4, 6.069e-4, 3.627e-5])$$

Матрица щума процесса

$$\mathrm{Q} = diag([1e-8,\ 1e-8, 1e-8,\ 9.243e-5,\ 1.329e-4,\ 2.172e-5])$$

Задание матриц для работы дискретного EKF

Вектор состояния: $\boldsymbol{x} = \begin{bmatrix} \vec{q} \\ \vec{w} \end{bmatrix}$ $\Lambda = [q_0, q_1, q_2, q_3]^T = \begin{bmatrix} q_0 \\ \vec{q} \end{bmatrix}$

$\vec{w}_{rel} = [w_x, w_y, w_z]^T$

Модель движения: $\Lambda_k = \left[I_{4x4} + \frac{1}{2} \Psi_{k-1} \Delta t \right] \Lambda_{k-1}$ $\Psi(\vec{w}_{rel}) = \begin{bmatrix} 0 & -w_x & -w_y & -w_z \\ w_x & 0 & w_z & -w_y \\ w_y & -w_z & 0 & w_x \\ w_z & w_y & -w_x & 0 \end{bmatrix}$

$\vec{w}_{rel_k} = \vec{w}_{rel_{k-1}}$

Модель измерений: $\boldsymbol{z_K} = \boldsymbol{H x_k} = \boldsymbol{h_{opt}}(\vec{\boldsymbol{q}})$ $\boldsymbol{H} = [\boldsymbol{I_3} \quad \boldsymbol{0_{3x3}}]$

Начальное значение матрицы ошибок: P(0) $= I_{6x6}$

Матрица щума измерений: $R = \begin{bmatrix} \sigma_{q1}^2 & 0 & 0 \\ 0 & \sigma_{q2}^2 & 0 \\ 0 & 0 & \sigma_{q3}^2 \end{bmatrix}$

, где σ_{q1}^2 = 6.123e-09 , σ_{q2}^2 = 1.318e-08, σ_{q3}^2 = 3.377e-07

Матрица щума процесса:

$$\text{Q} = diag([1e-8, 4.99e-12, 1.e-8, 6.28e-06, 4.55e-06, 2.47e-06])$$

Линеаризация уравнений движения:

$$\boldsymbol{F} = \begin{bmatrix} \partial\vec{q}_k/\partial\boldsymbol{x} \\ \partial\vec{w}_{rel_k}/\partial\boldsymbol{x} \end{bmatrix}$$

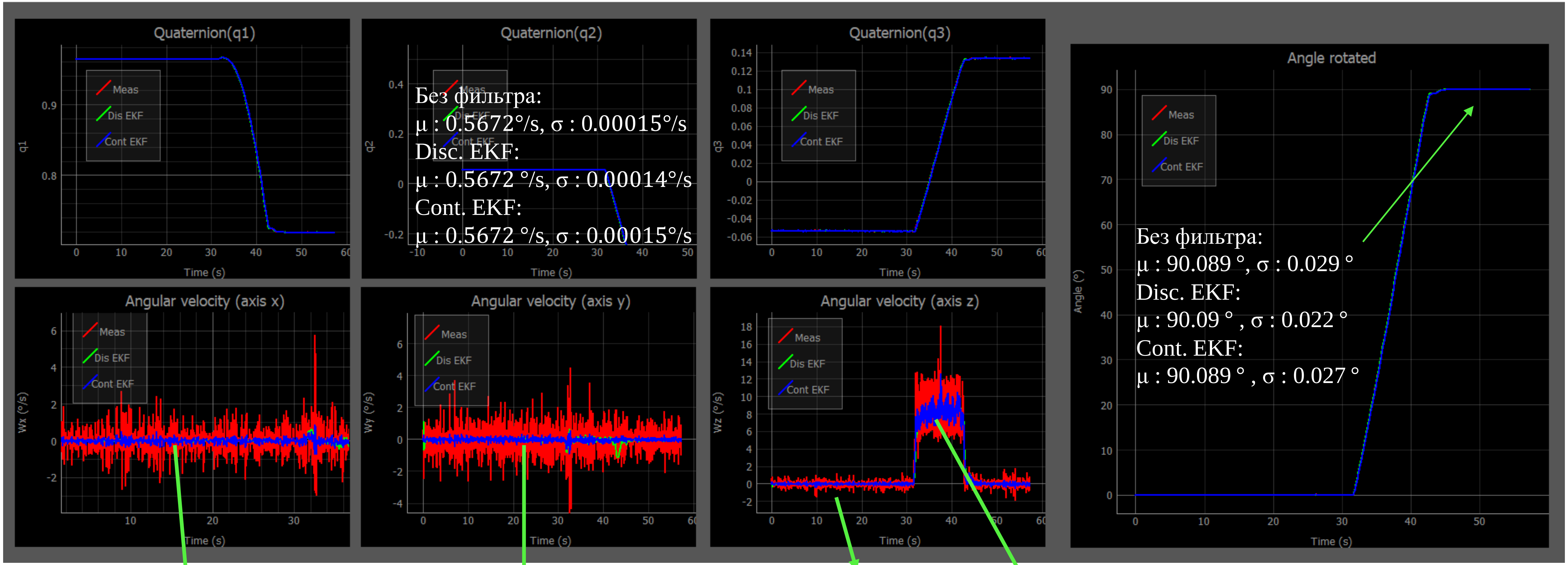
$$\frac{\partial\vec{q}_k}{\partial\boldsymbol{x}} = \begin{bmatrix} \frac{\partial\vec{q}_k}{\partial\vec{q}} & \frac{\partial\vec{q}_k}{\partial\vec{w}_{rel}} \end{bmatrix}$$

$$\frac{\partial\vec{q}_k}{\partial\vec{q}} = \begin{bmatrix} \frac{\partial q_1}{\partial\vec{q}} \\ \frac{\partial q_2}{\partial\vec{q}} \\ \frac{\partial q_3}{\partial\vec{q}} \end{bmatrix} = \begin{bmatrix} 1 - \frac{q_1}{q_0}\frac{w_x\Delta t}{2} & -\frac{q_2}{q_0}\frac{w_x\Delta t}{2} + \frac{w_z\Delta t}{2} & -\frac{q_3}{q_0}\frac{w_x\Delta t}{2} - \frac{w_y\Delta t}{2} \\ -\frac{q_1}{q_0}\frac{w_y\Delta t}{2} - \frac{w_z\Delta t}{2} & 1 - \frac{q_2}{q_0}\frac{w_y\Delta t}{2} & -\frac{q_3}{q_0}\frac{w_y\Delta t}{2} + \frac{w_x\Delta t}{2} \\ -\frac{q_1}{q_0}\frac{w_z\Delta t}{2} - \frac{w_y\Delta t}{2} & -\frac{q_2}{q_0}\frac{w_z\Delta t}{2} - \frac{w_x\Delta t}{2} & 1 - \frac{q_3}{q_0}\frac{w_z\Delta t}{2} \end{bmatrix}$$

$$\frac{\partial\vec{q}_k}{\partial\vec{w}_{rel}} = \begin{bmatrix} q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix}$$

$$\frac{\partial\vec{w}_{rel_k}}{\partial\boldsymbol{x}} = [\mathbf{0}_{3x3} \quad \boldsymbol{I}_3]$$

Опыт II : Поворот на 90 градусов



Без фильтра:
μ : 0.5672°/s, σ : 0.00015°/s
Disc. EKF:
μ : 0.5672 °/s, σ : 0.00014°/s
Cont. EKF:
μ : 0.5672 °/s, σ : 0.00015°/s

Без фильтра:
μ : 90.089 °, σ : 0.029 °
Disc. EKF:
μ : 90.09 ° , σ : 0.022 °
Cont. EKF:
μ : 90.089 ° , σ : 0.027 °

Без фильтра:
μ : 0.004°/s, σ : 0.791°/s
Disc. EKF: (90%)
μ : 0.004°/s, σ : 0.078°/s
Cont. EKF: (81.7%)
μ : 0.004°/s, σ : 0.144°/s

Без фильтра:
μ : 0.001°/s, σ : 1.02°/s
Disc. EKF: (90%)
μ : -0.002°/s, σ : 0.08°/s
Cont. EKF: (86%)
μ : -0.001°/s, σ : 0.13°/s

Без фильтра:
μ : -0.003°/s , σ : 0.412°/s
Disc. EKF: (89.1%)
μ : -0.002°/s , σ : 0.043°/s
Cont. EKF: (70%)
μ : -0.002°/s , σ : 0.121°/s

Без фильтра:
μ : 7.97°/s , σ : 2.76°/s
Disc. EKF: (80%)
μ : 7.89°/s , σ : 0.53°/s
Cont. EKF: (65.7%)
μ : 7.94°/s , σ : 0.95°/s

Заключение

- Ориентацию и угловую скорость объекта можно было оценить с помощью камеры без необходимости дополнительных датчиков. Результаты показали повышение точности до 80% при оценке угловой скорости.
- Что касается ориентации, которая определяется кватернионами, не было значительного улучшения, это потому, что измерение уже является точным.
- Модели наблюдений, основанные на алгоритме калибровки, позволяют получать точные результаты в матрице поборота, что позволяет получать хорошие результаты в приложении EKF при оценке угловой скорости.
- При использовании модели наблюдения, основанной на модели измерения h_{X_i}, никакого результата не было получено, это связано с его высокой нелинейностью. Это потребует большего анализа для будущей работы

Литература

[1] R. Y. Tsai, “*A Versatile Camera Calibration Techniaue for High-Accuracy 3D Machine Vision Metrology Using Off-the-shelf TV Cameras and Lenses*, IEEE Journal on Robotics and Automation, 1987, Vol. 3, No. 4, pp. 323 – 344.

[2] Z. Zhang. Flexible Camera Calibration By Viewing a Plane From Unknown Orientations, Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, Greece, September 20-27,1999

[3] A.A.Boguslavsky, V.V.Sazonov, S.M.Sokolov, A.I.Smirnov, Kh.S.Saigiraev. *Automatic vision-based monitoring of the spacecraft docking approach with the International Space Station*, Proceedings of the First International Conference on Informatics in Control, Automation and Robotics, 2004, pp. 79-86

[4] Иванов Д. С., Карпенко С.О., Овчинников М. Ю., Сакович М.А. *Определение относительного движения спутников при их разделении по результатам обработки видеоизображения*, Препринт ИПМ им. М.В. Келдыша, 2012, № 57, 24 с.

[5] D. C. Brown, *Decentering Distortion of Lenses*, *Photometric Engineering* 32(3), 1966, pp. 444–462

[6] [7]Bouguet, J. Y. Camera Calibration Toolbox for Matlab, 2015, Available: http://www.vision.caltech.edu/bouguetj/calib_doc/ (дата обращения: 20.06.220)

[7] F. J. Romero-Ramirez, R. Muñoz-Salinas, R. Medina-Carnicer. *Speeded up detection of squared fiducial markers*, Image and Vision Computing, 2018, vol 76, pages 38-47

Спасибо за внимание