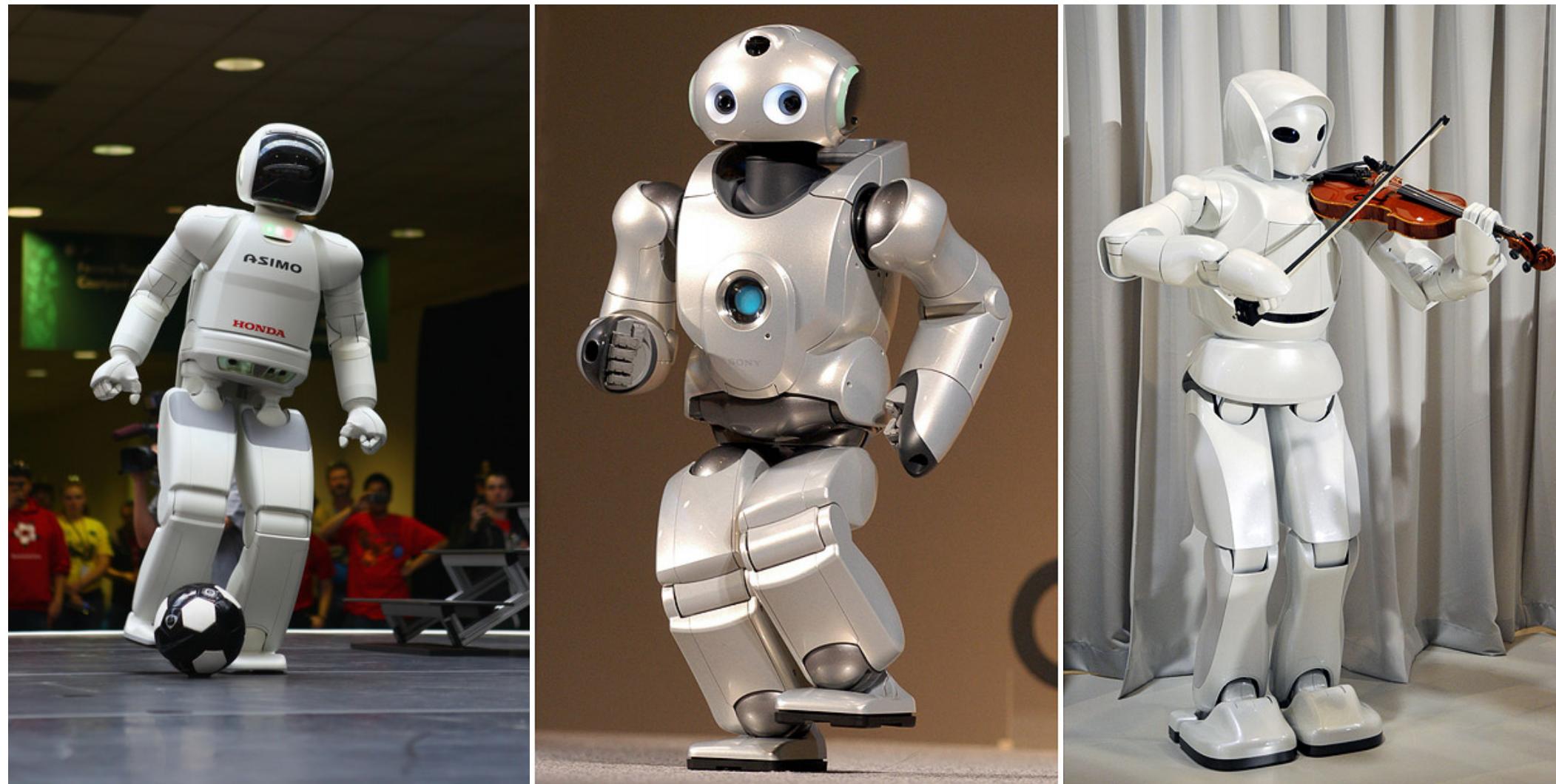


Распознавание цветных объектов, обратная кинематика и модель камеры-обскуры для управления рукой-манипулятором

XI-ая Международная Азиатская Школа-семинар
«Проблемы оптимизации сложных систем»

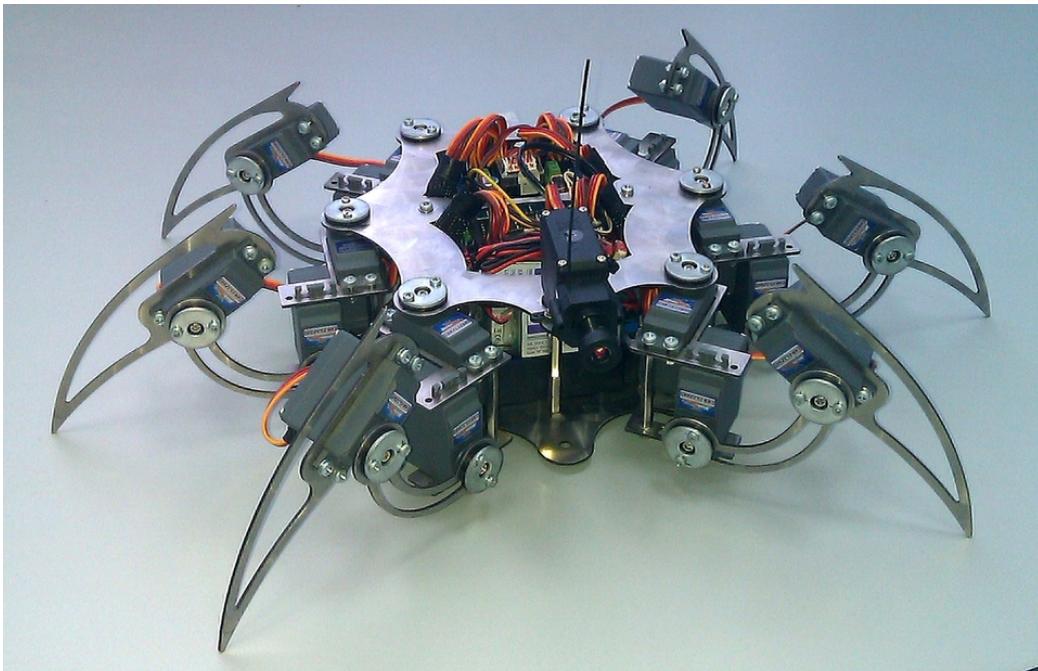
27 июля — 7 августа 2015 года
Кыргызстан, Иссык-Куль, г. Чолпон-Ата

≡ Вступление

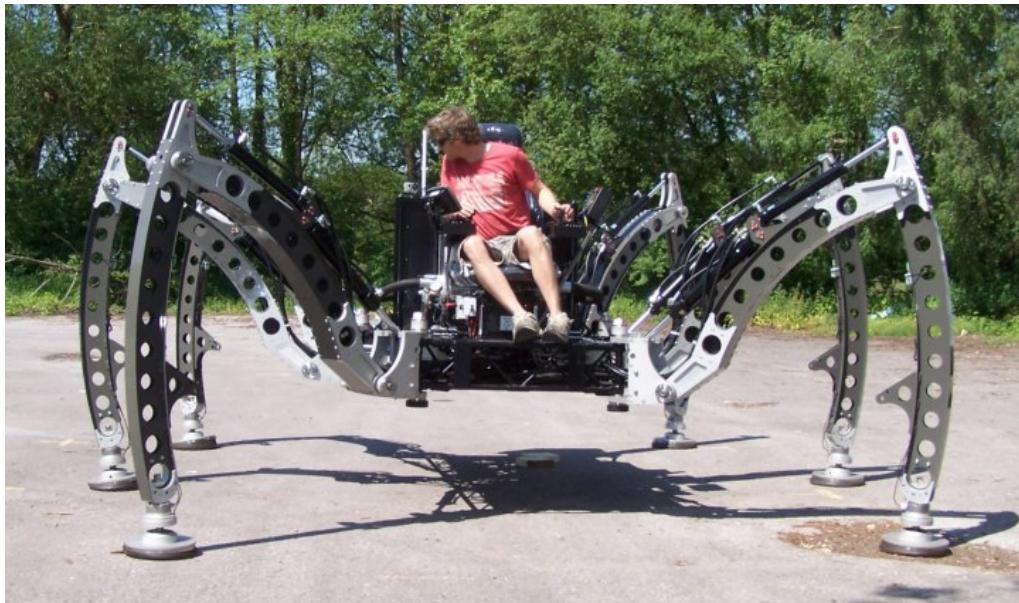


- Роботы гуманоиды;
- Для управления используются алгоритмы машинного обучения.

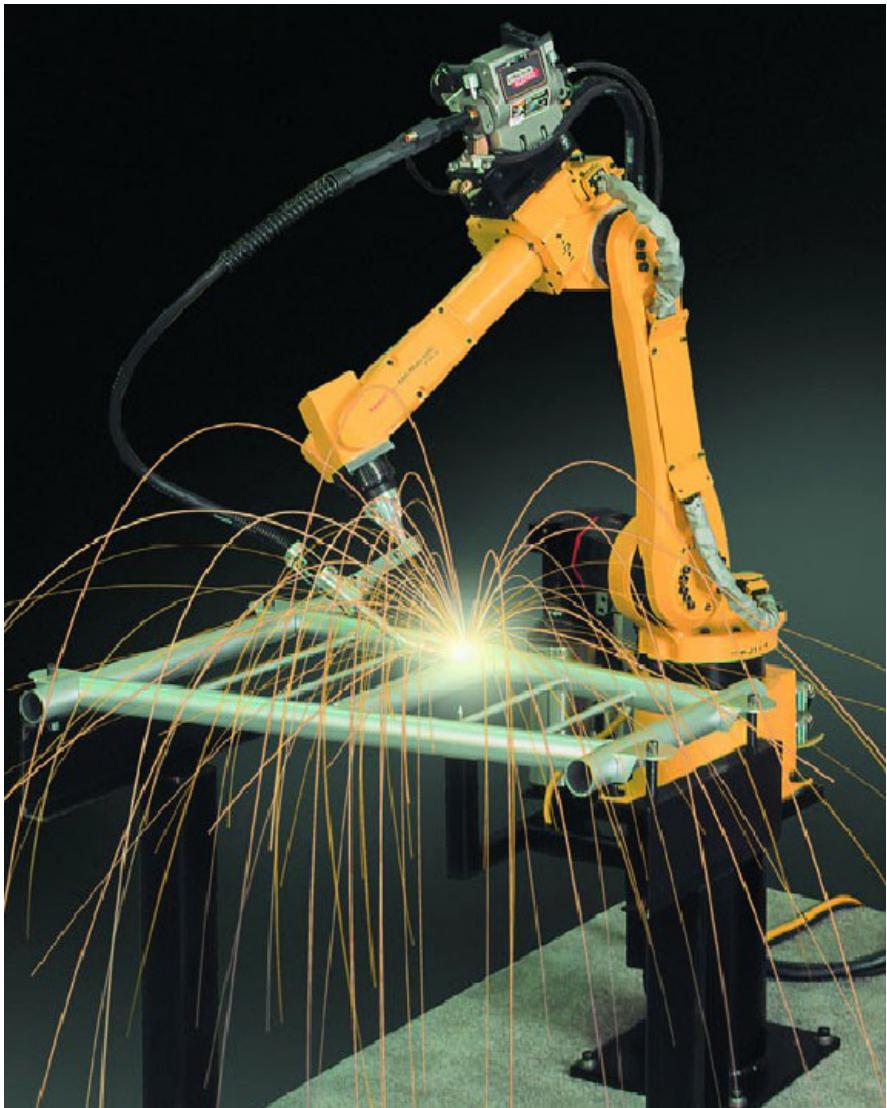
≡ Вступление



- Шестиногие роботы-манипуляторы (hexapod robots);
- Управление осуществляется напрямую человеком или при помощи точных алгоритмов.



≡ Рука-манипулятор



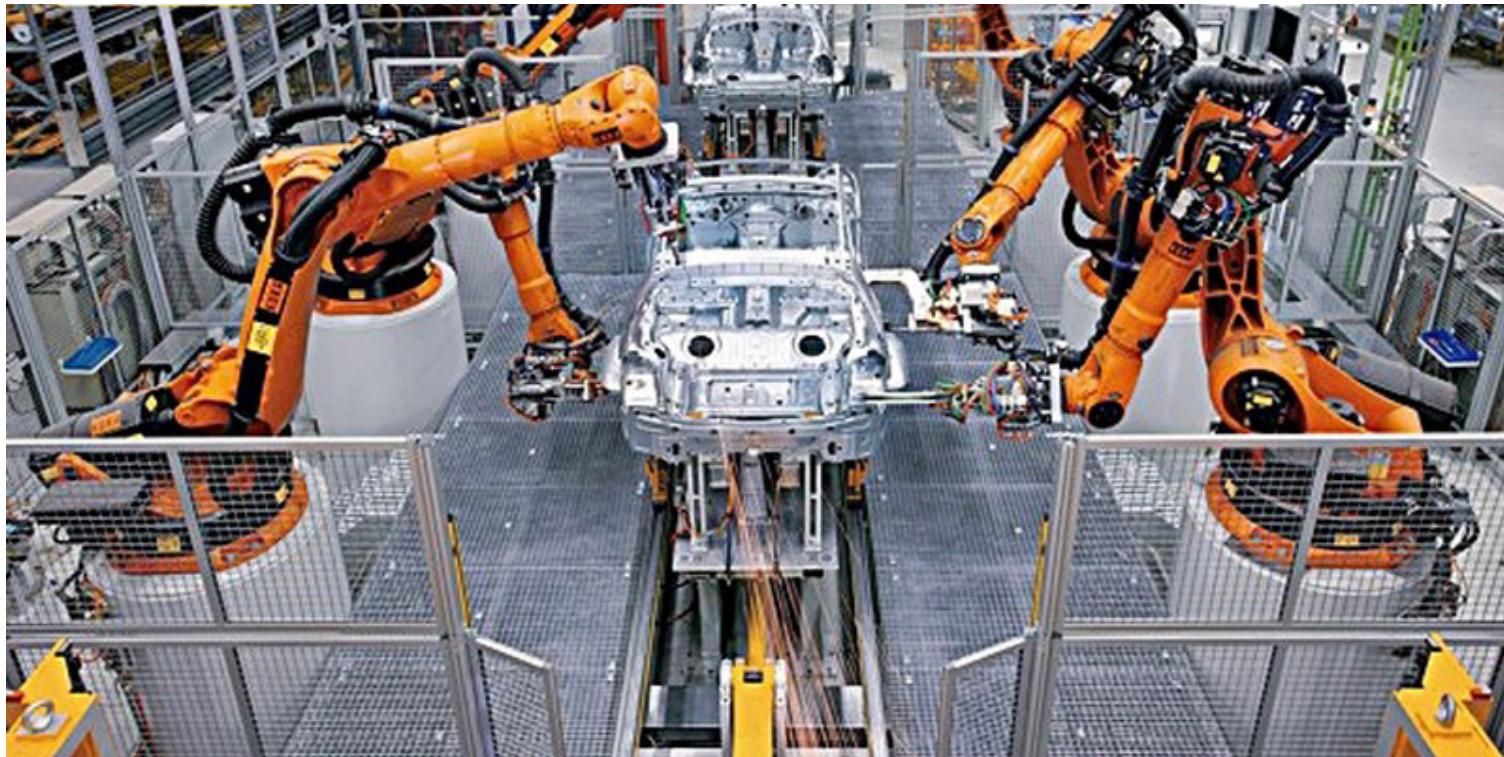
Сварка электрической дугой



За работой возле печи

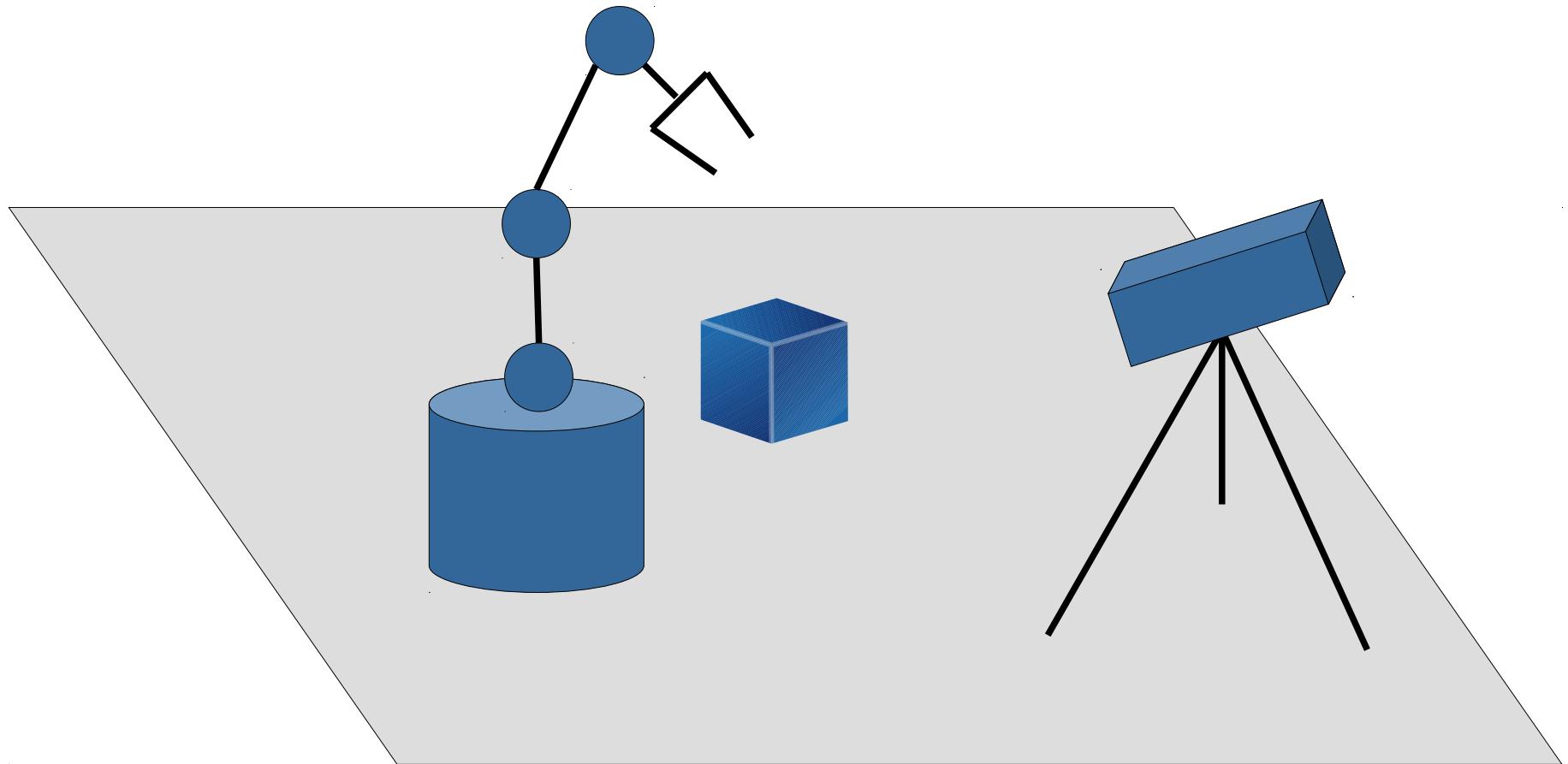
≡ Рука-манипулятор

- Таким образом, роботы-манипуляторы заменяют труд человека там, где его нахождение опасно (высокий радиоактивный фон, высокая температура), или же точность работы человека недостаточна.



Применение в автомобильной промышленности

≡ Постановка задачи

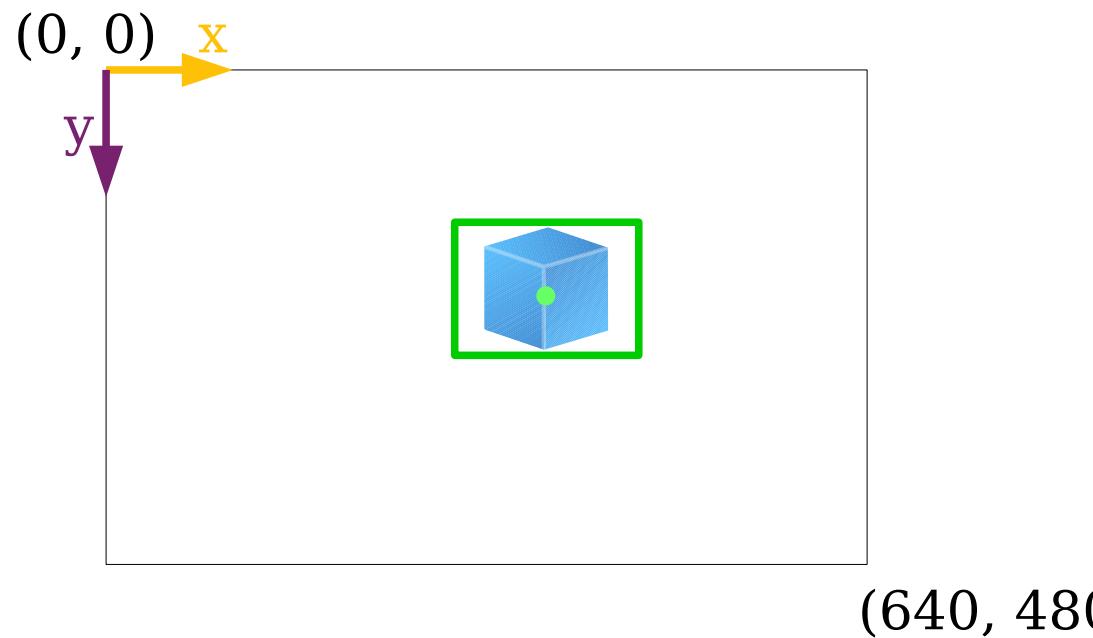


≡ Постановка задачи

1. Калибровка камеры;
2. Распознавание объекта на кадре с камеры;
3. Проецирование координат объекта в реальный мир;
4. Решение задачи обратной кинематики;
5. Разработка низкоуровневого ПО для микроконтроллера.

≡ Распознавание объекта

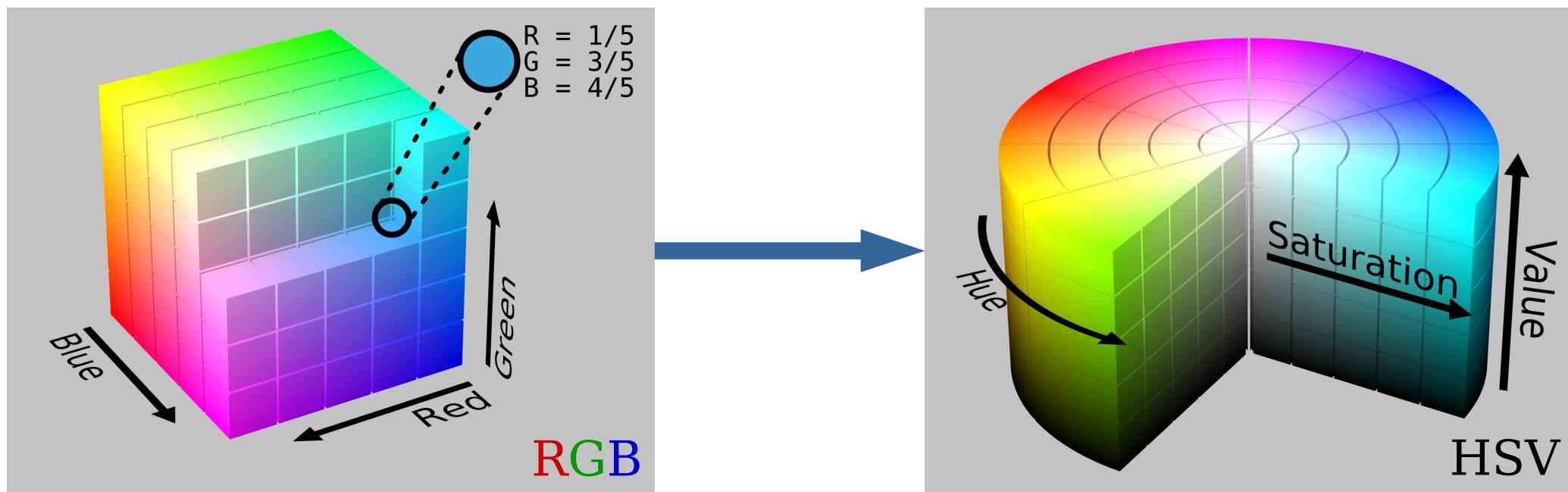
Цель: выделение объекта на входном кадре с камеры обрамляющим прямоугольником для определения координат приблизительного центра масс объекта относительно системы координат двумерного изображения.



- Используется для определения объекта заранее заданного цвета;
- Устойчиво к наличию на поверхности объекта регионов с цветом, отличным от искомого.

≡ Распознавание объекта

1. Трехканальная матрица исходного изображения кадра переводится из цветового пространства RGB (Red, Green, Blue) в HSV (Hue, Saturation, Value), так как это позволяет наиболее эффективно задавать диапазон цветов необходимого оттенка;

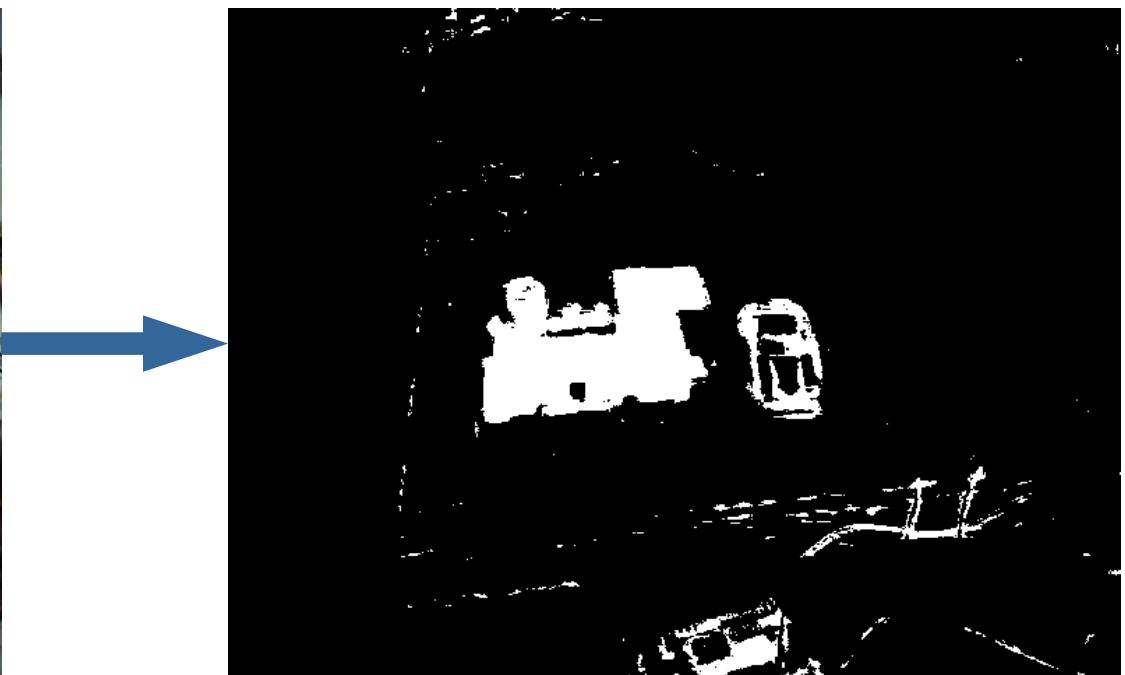
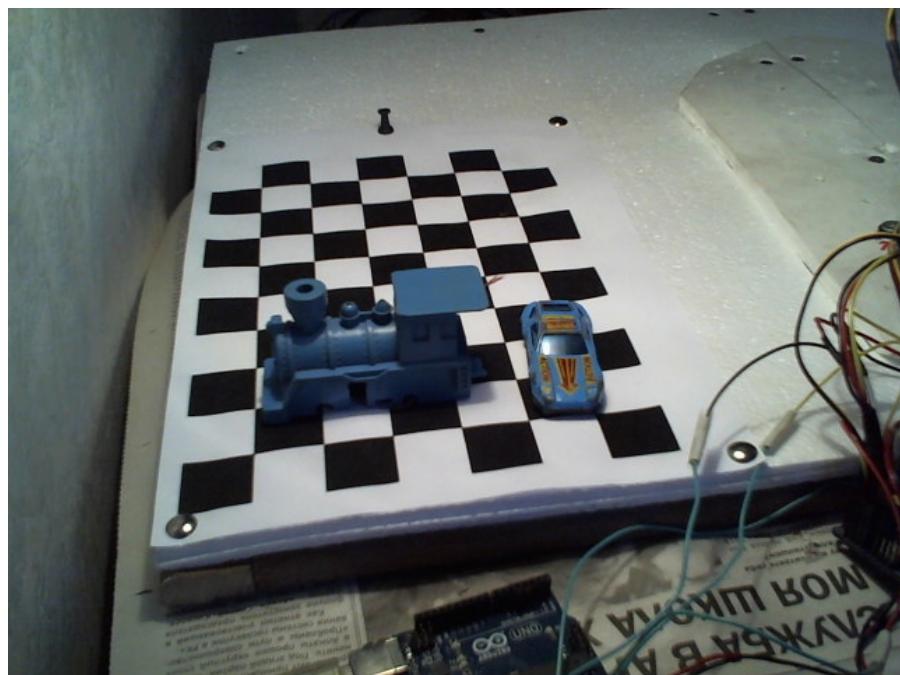
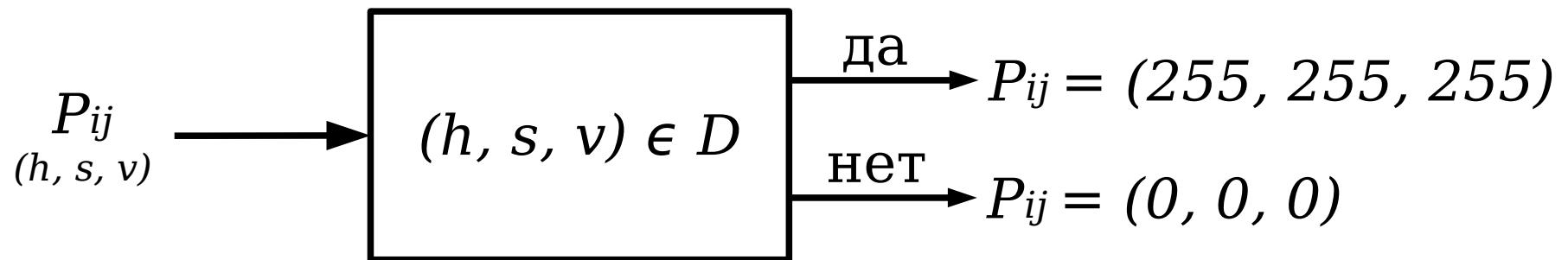


К примеру, диапазон D для цветов синего оттенка:

$$D = \text{[light blue square]} (200, 30, 90) — \text{[dark blue square]} (250, 100, 100)$$

≡ Распознавание объекта

2. Изображение переводится в бинарное путем проверки вхождения каждого пикселя P_{ij} в диапазон.

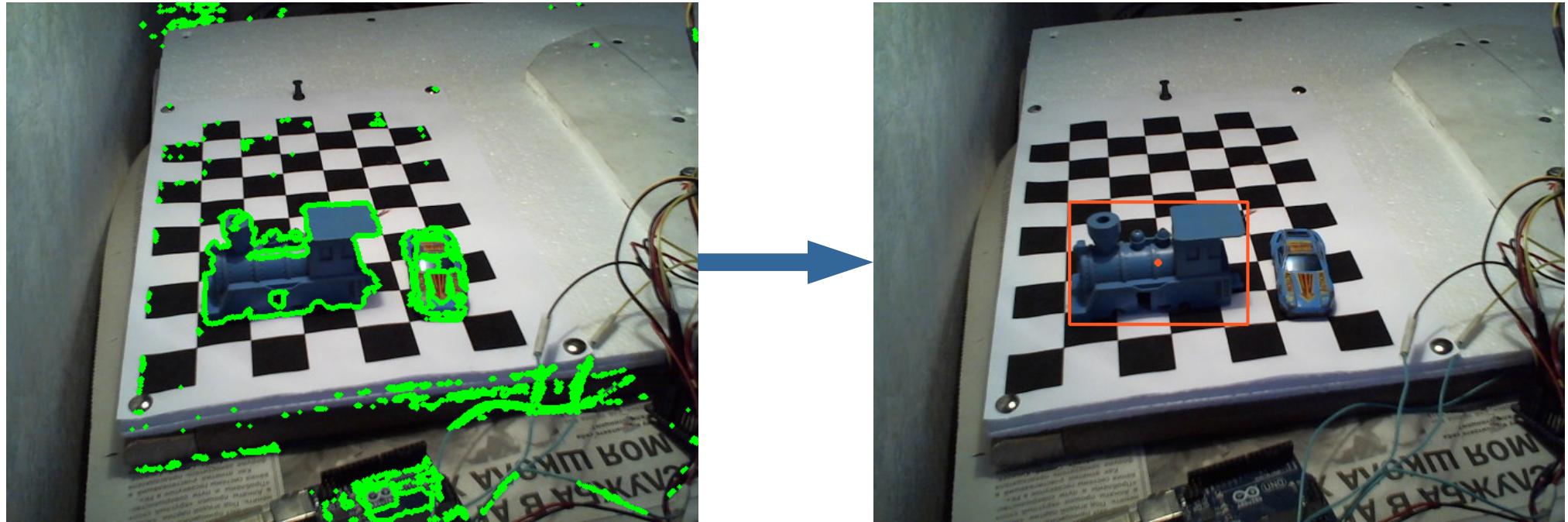


☰ Распознавание объекта

3. Применяется алгоритм поиска контуров в полученной маске^[1], реализованный в библиотеке OpenCV.

Алгоритм возвращает массив обнаруженных контуров, из которых отбирается контур с наибольшей площадью.

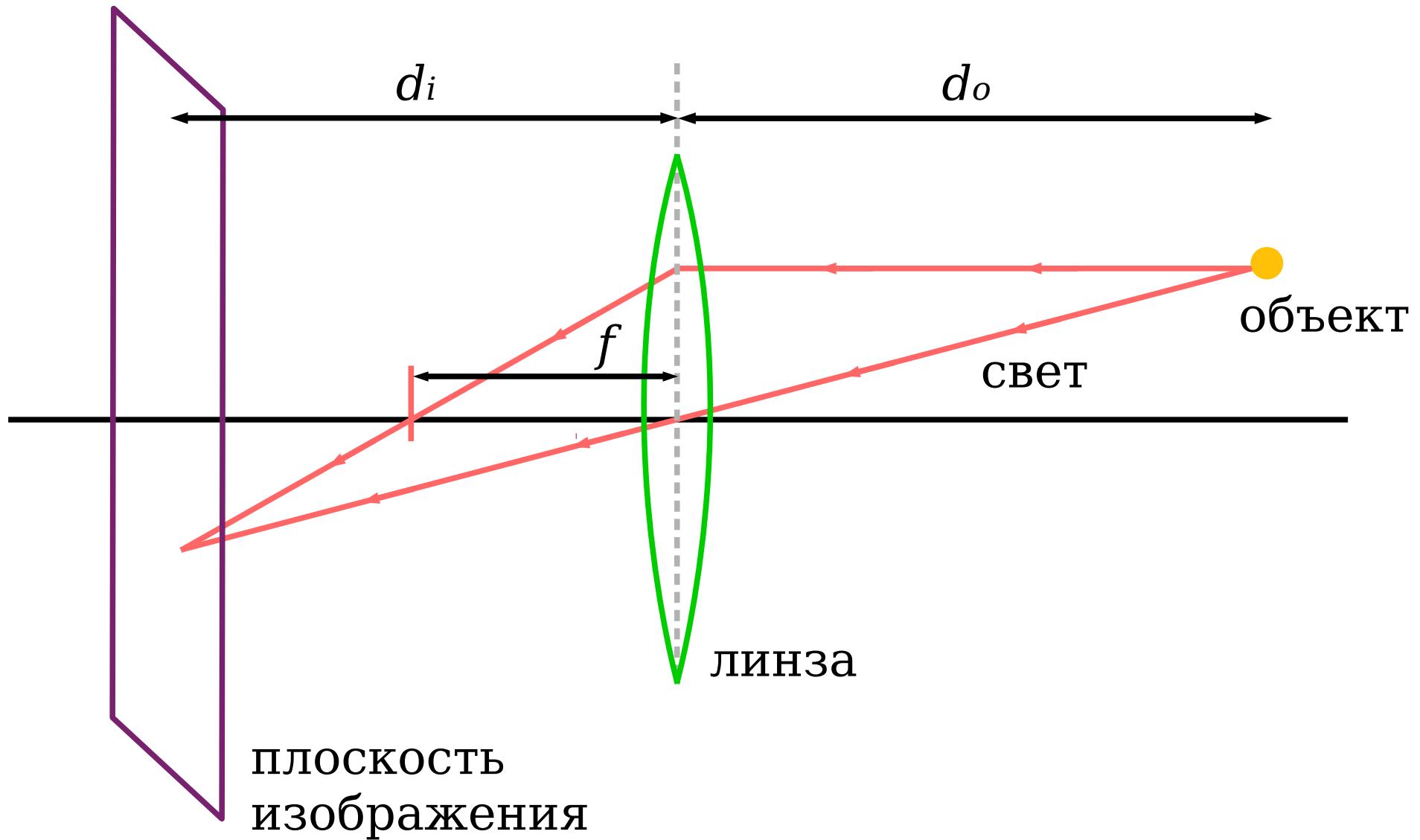
4. Крайние точки контура используются для построения обрамляющего прямоугольника, а точка пересечения диагоналей воспринимается как центр масс объекта.



[1] Topological Structural Analysis of Digitized Binary Images by Border Following, Satoshi Suzuki © M.R.R. 2015

≡ Фотокамера

- Путь лучей в камере через линзу в объективе.



≡ Формула линзы

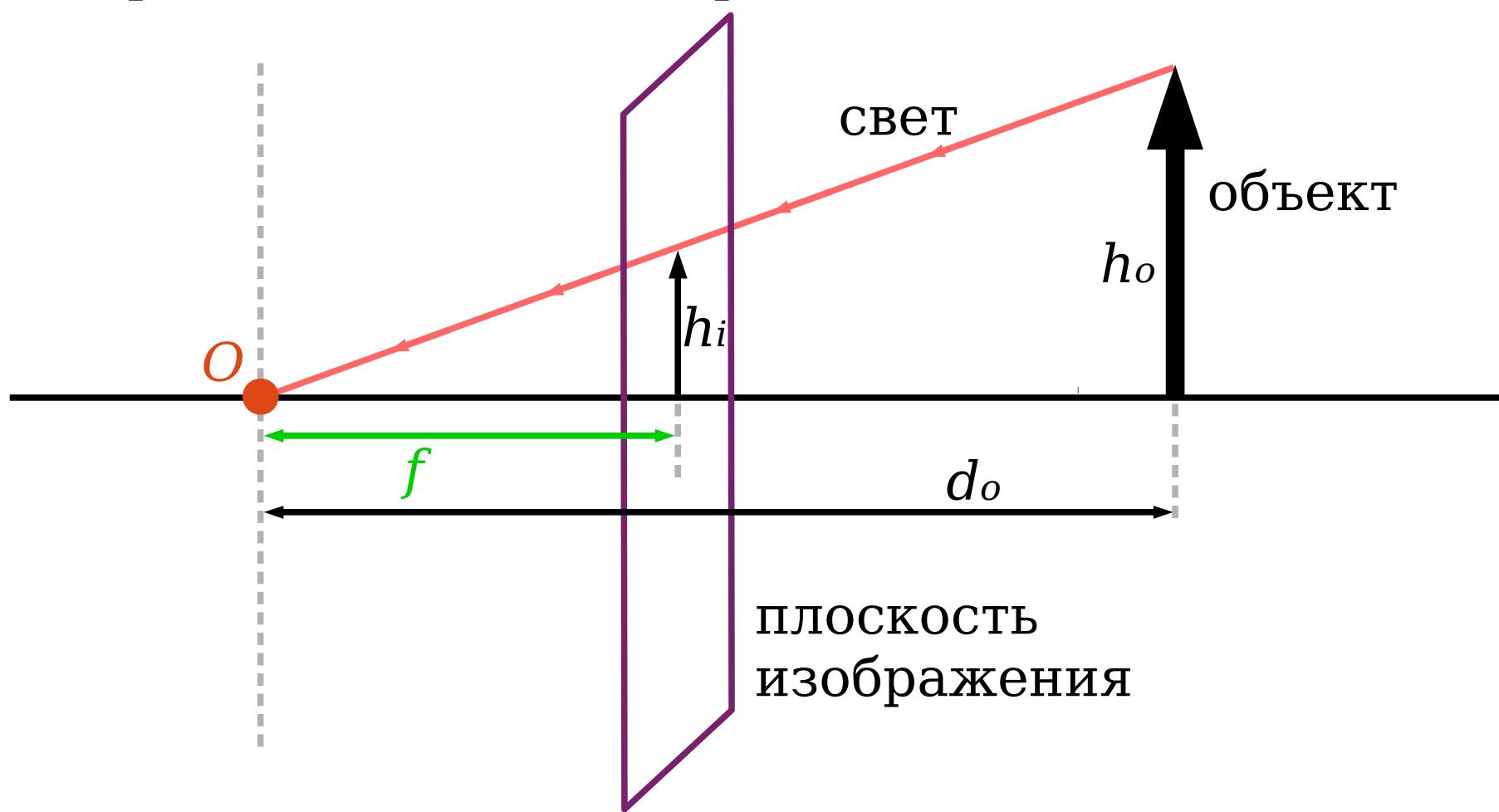
$$\frac{1}{f} = \frac{1}{d_o} + \frac{1}{d_i} \quad (1)$$

- d_o — расстояние от объекта до линзы;
- d_i — расстояние от линзы до изображения;
- f — фокусное расстояние линзы.

≡ Модель камеры-обскуры

Упрощенная модель системы в компьютерном зрении:

- Пренебрегается эффект линзы;
- $d_o \gg d_i$. Тогда можно предположить, что $d_i = f$;
- Изображение мнимое и прямое.

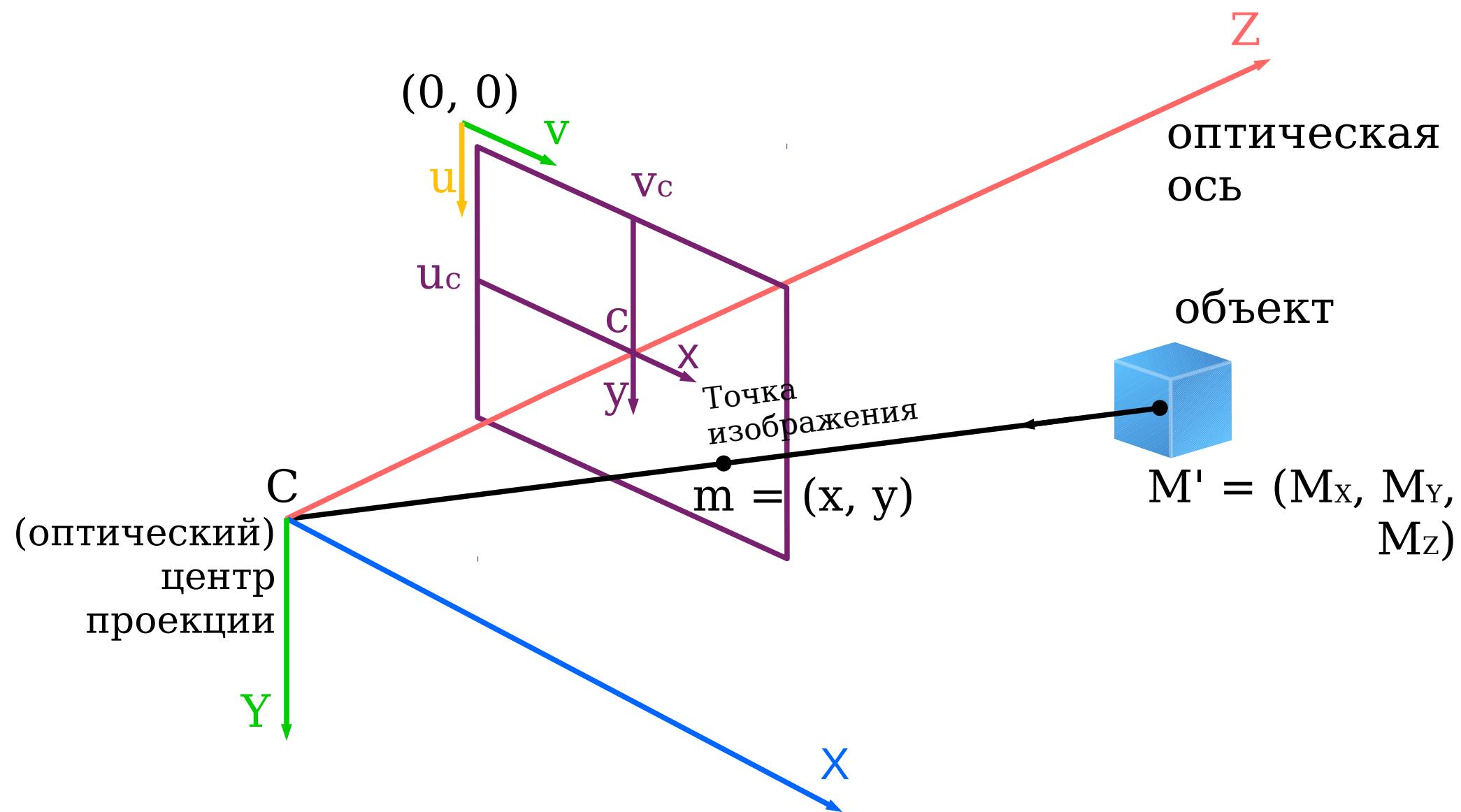


≡ Модель камеры-обскуры

$$h_i = f \frac{h_o}{d_o} \quad (2)$$

- h_i — высота изображения объекта;
- d_o — расстояние от объекта до оптического центра;
- h_o — высота наблюдаемого объекта;
- f — фокусное расстояние линзы.

≡ Координатные преобразования



≡ Координатные преобразования

$$x = f \frac{M_X}{M_Z}; y = f \frac{M_Y}{M_Z} \quad (3)$$

$$\begin{bmatrix} sx \\ sy \\ s \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} M_X \\ M_Y \\ M_Z \\ 1 \end{bmatrix} \quad (4)$$

- x, y — координаты точки на изображении в системе сху;
- M_x, M_y, M_z — координаты объекта в системе CXYZ;
- s — ненулевой скалярный множитель;
- f — фокусное расстояние линзы.

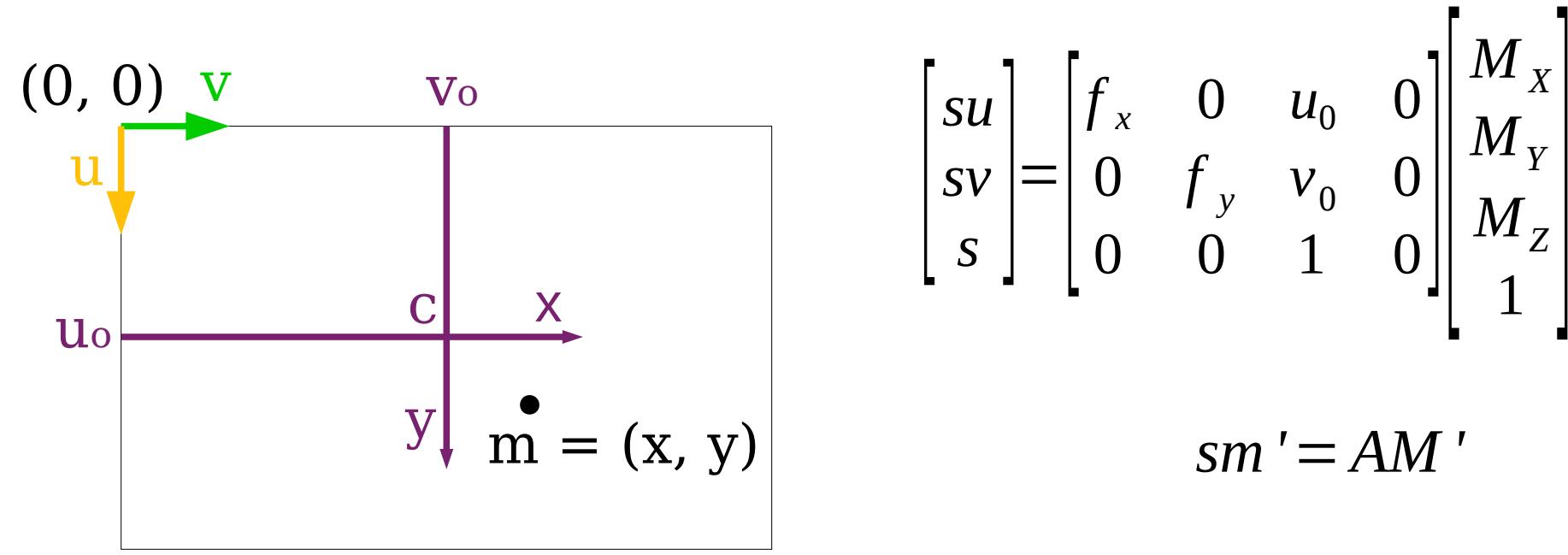
≡ Координатные преобразования

$$u = u_0 + \frac{x}{\text{ширина пикселя}}; v = v_0 + \frac{y}{\text{высота пикселя}} \quad (5)$$

$$(3) \rightarrow (5): M_Z u = M_Z u_0 + f_x M_X; M_Z v = M_Z v_0 + f_y M_Y \quad (6)$$

Где:

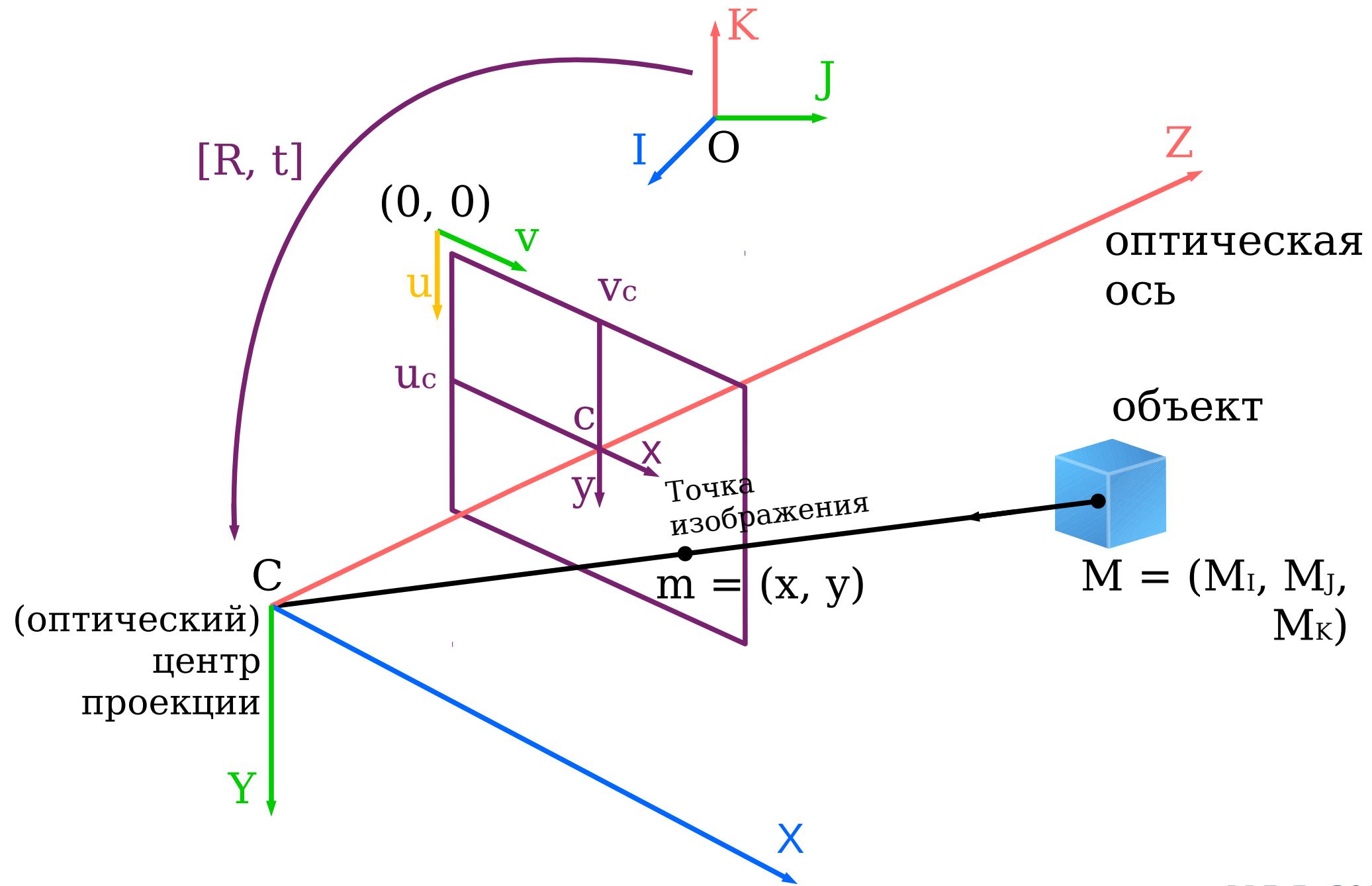
$$f_x = \frac{f}{\text{ширина пикселя}}; f_y = \frac{f}{\text{высота пикселя}} \quad (7)$$



$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} M_X \\ M_Y \\ M_Z \\ 1 \end{bmatrix} \quad (8)$$

$$sm' = AM' \quad (9)$$

≡ Координатные преобразования



≡ Координатные преобразования

$$M = \begin{bmatrix} M_I \\ M_J \\ M_K \\ 1 \end{bmatrix} \xrightarrow{[R, t]} M' = \begin{bmatrix} M_X \\ M_Y \\ M_Z \end{bmatrix} \quad (10)$$

$$[R|t] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \quad (11)$$

$$M' = [R|t]M \quad (12)$$

≡ Координатные преобразования

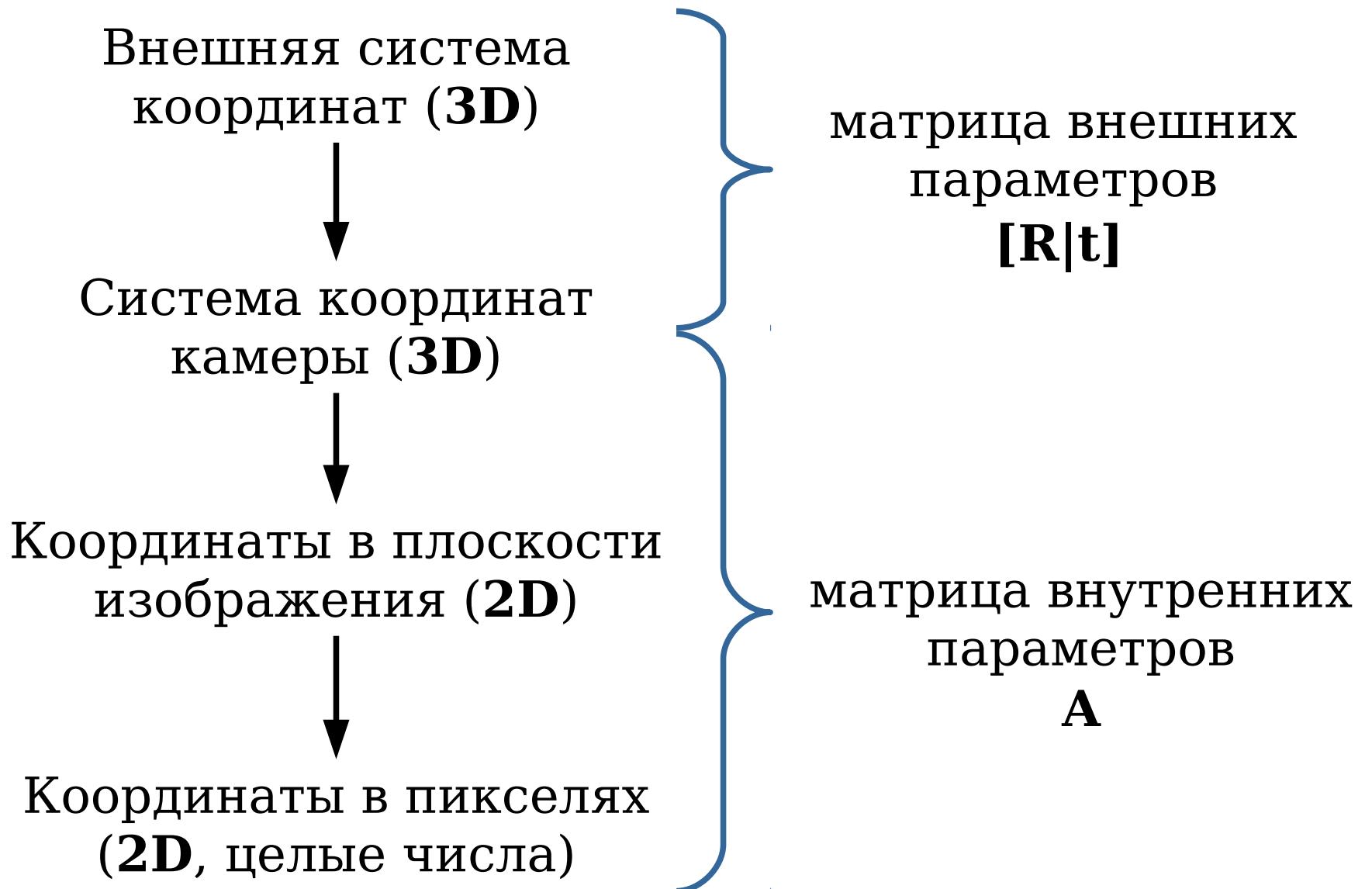
- Фундаментальное уравнение, устанавливающее связь между 2D изображением и реальным миром.

$$sm' = A[R|t]M \quad (13)$$

$$S \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} M_I \\ M_J \\ M_K \\ 1 \end{bmatrix} \quad (14)$$

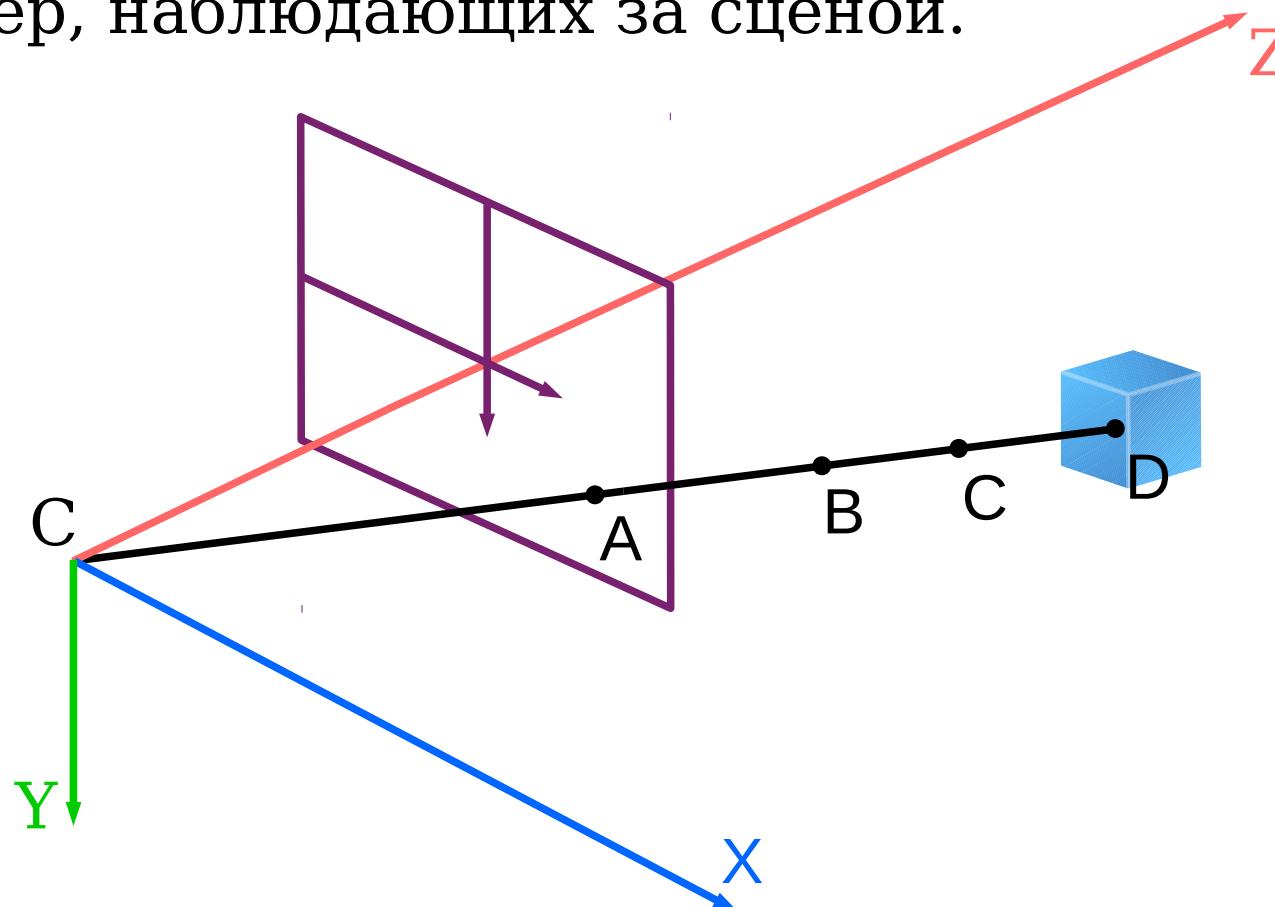
- m' — вектор координат в системе 2D изображения;
- A — матрица внутренних параметров камеры;
- $[R|t]$ — матрица внешних параметров камеры;
- M — вектор координат объекта в реальном мире.

≡ Координатные преобразования

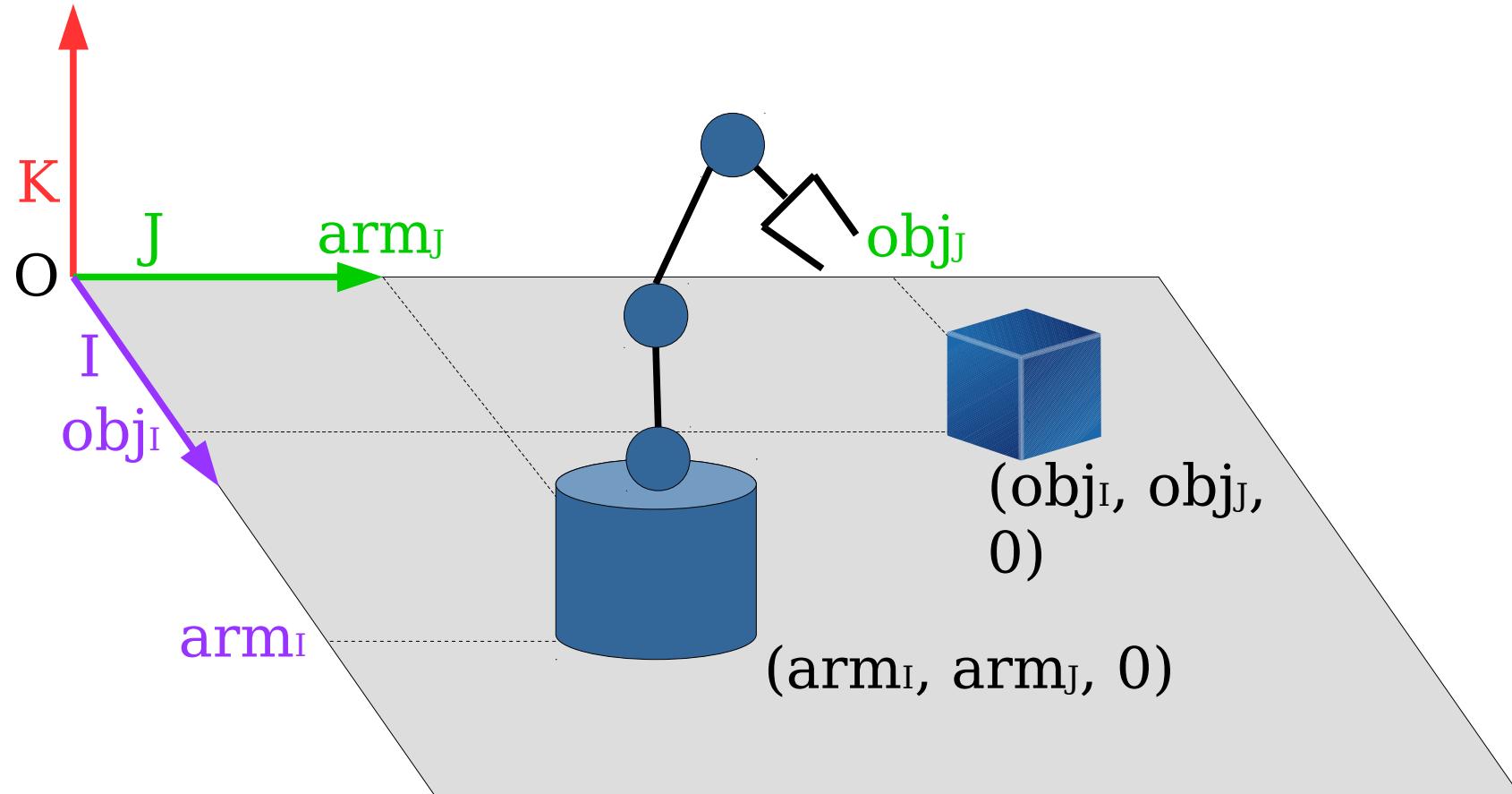


≡ Координатные преобразования

- Фундаментальное уравнение (14) позволяет проецировать точки из реального мира на плоскость изображения имея в распоряжении всего одну камеру;
- Для обратной задачи необходимо либо знать заранее одну из 3D координат объекта, или получать данные от двух камер, наблюдающих за сценой.



≡ Координатные преобразования



≡ Координатные преобразования

$$M_K=0 \rightarrow (5): \quad sm' = A \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} M_I \\ M_J \\ 0 \\ 1 \end{bmatrix} \quad (15)$$

$$sm' = A \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix} \begin{bmatrix} M_I \\ M_J \\ 1 \end{bmatrix}; \quad \begin{bmatrix} M_I \\ M_J \\ 1 \end{bmatrix} = s \left(A \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix} \right)^{-1} m' \quad (16)$$

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \left(A \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix} \right)^{-1} m'; \quad \begin{bmatrix} M_I \\ M_J \\ 1 \end{bmatrix} = \begin{bmatrix} se_1 \\ se_2 \\ se_3 \end{bmatrix} \quad (17)$$

$$s = \frac{1}{e_3}; M_I = \frac{e_1}{e_3}; M_J = \frac{e_2}{e_3} \quad (18)$$

≡ Калибровка камеры

Калибровка камеры позволяет определить:

1. Внутренние параметры камеры:

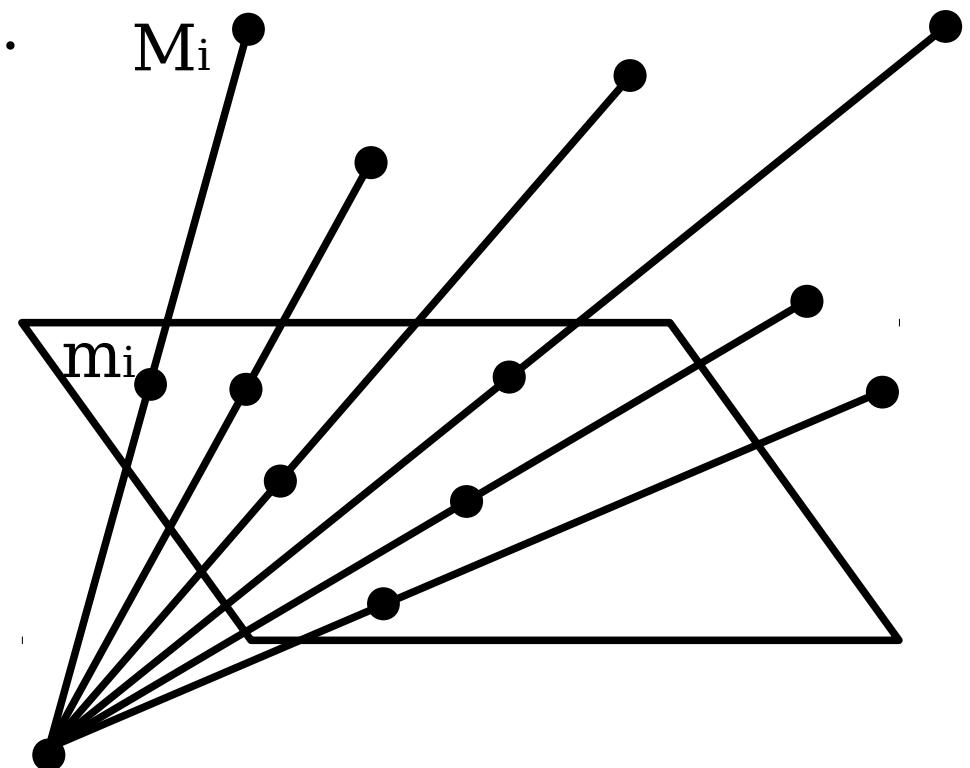
- Фокусное расстояние линзы f ;
- Координаты главной точки изображения (u_o, v_o) ;
- Ширину и высоту пикселя.

2. Внешние параметры:

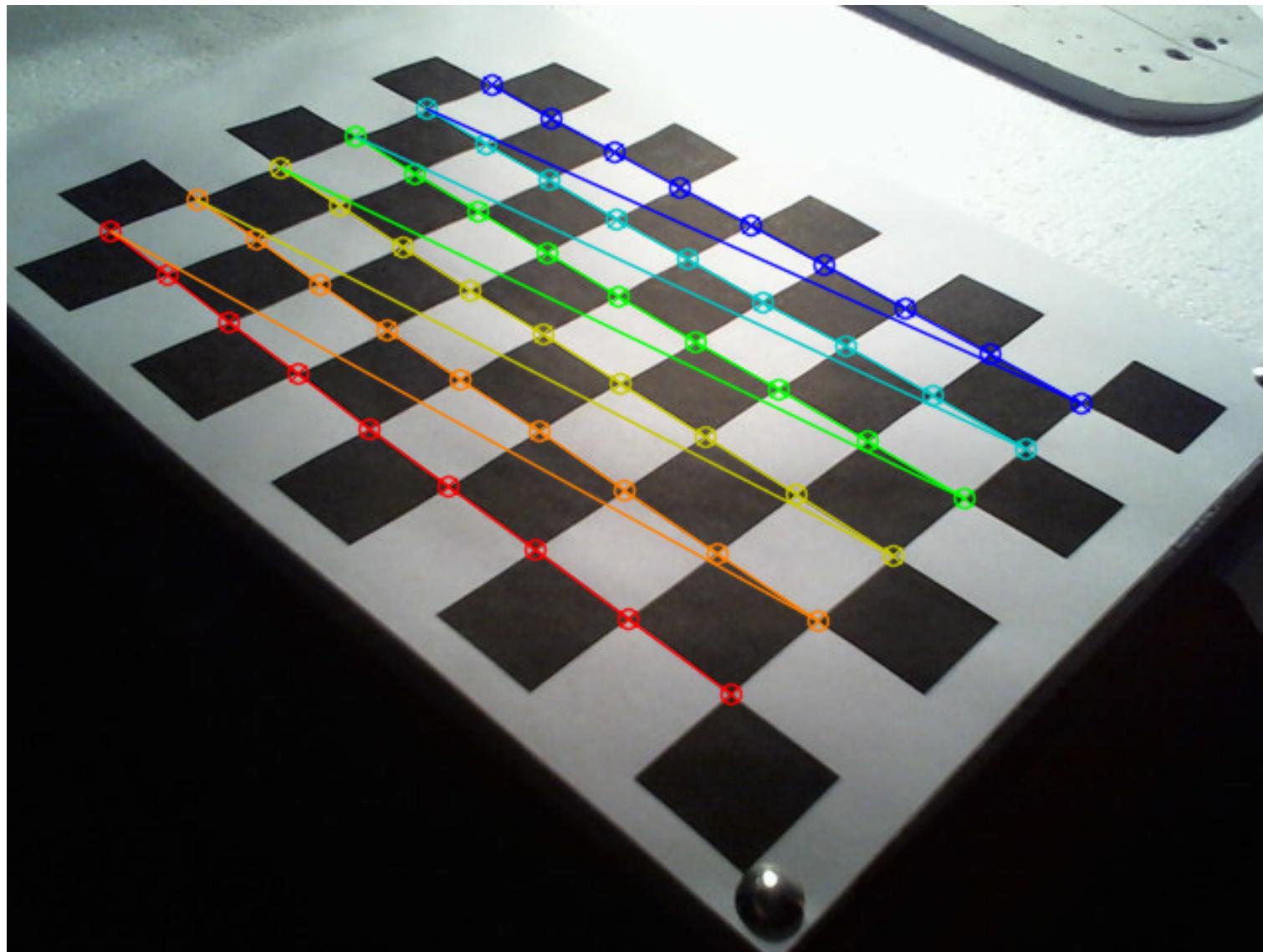
- Вектор поворота;
- Вектор перемещения.

Всего 11 параметров.

Калибровка требует как минимум 6 соответствий точек на изображении и в пространстве, так как каждое соответствие пространственной точки M_i точке на изображении m_i порождает 2 независимых уравнения.



≡ Реализация в OpenCV



Определение углов пересечения шахматной доски

≡ Реализация в OpenCV



Результат применения алгоритма

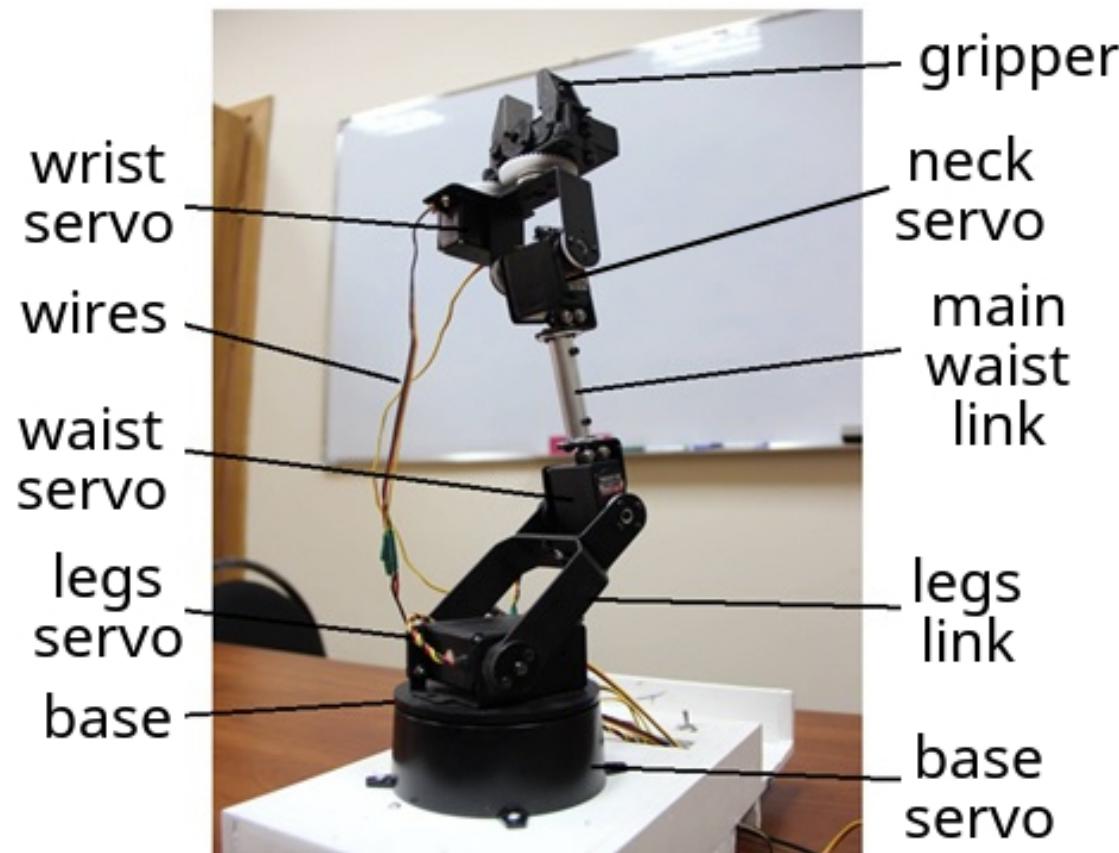
≡ Кинематика

На данном этапе нам известны:

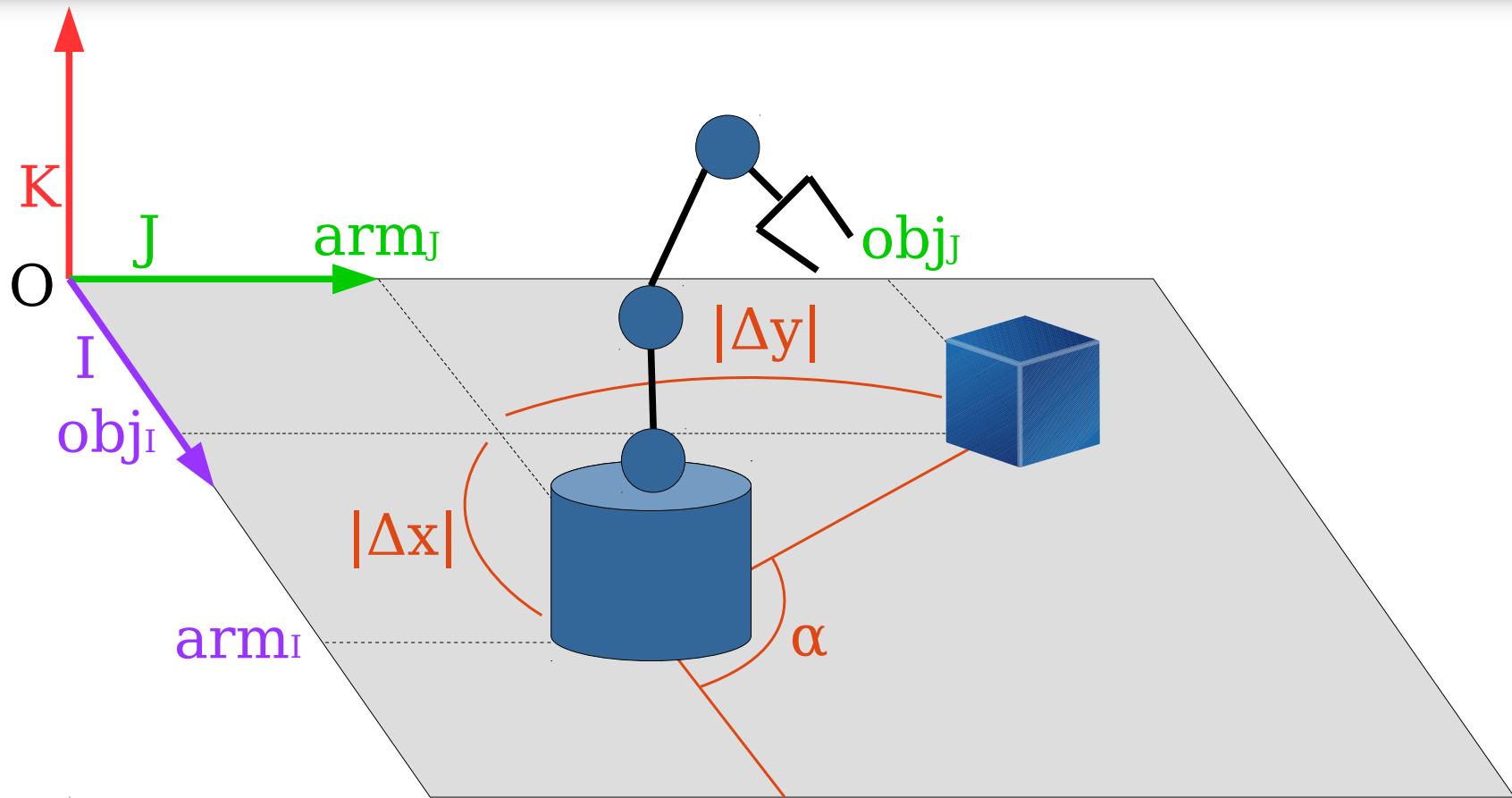
- Координаты объекта (M_I, M_J);
- Координаты основания руки-манипулятора;
- Параметры руки-манипулятора.

Необходимо
определить:

- Соответствующие углы поворота сервоприводов в суставах руки, способных обеспечить достижение хватом объекта.



≡ Кинематика

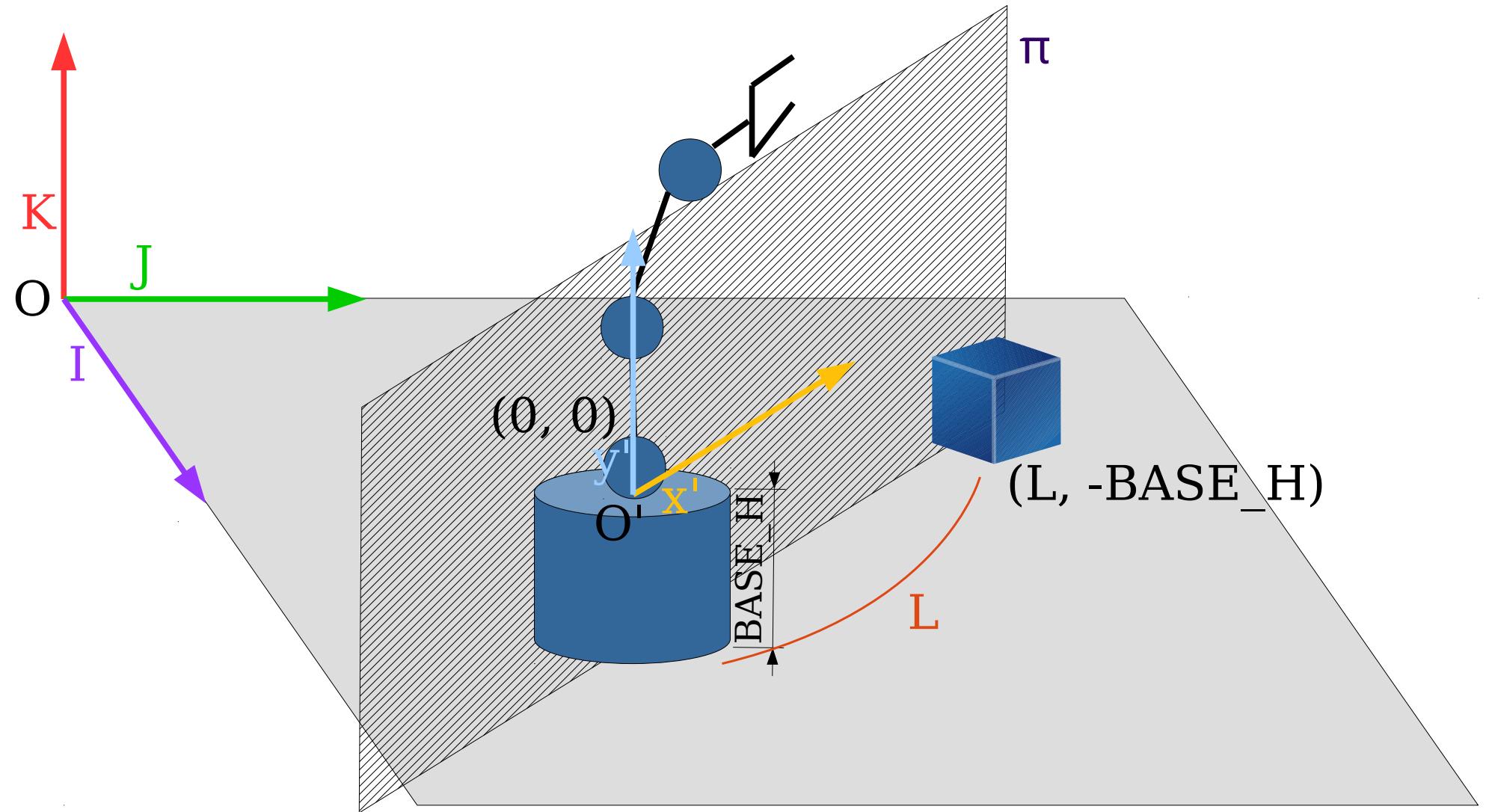


$$\Delta x = obj_I - arm_I; \quad \Delta y = obj_J - arm_J \quad (19)$$

$$L = \sqrt{\Delta x^2 + \Delta y^2}; \quad \alpha_d = \arccos \frac{\Delta x}{L} rad \quad (20)$$

$$\alpha = \alpha_d \frac{180^\circ}{\pi rad} \quad (21)$$

≡ Кинематика



Проблема переходит из пространства в плоскость

≡ Прямая кинематика

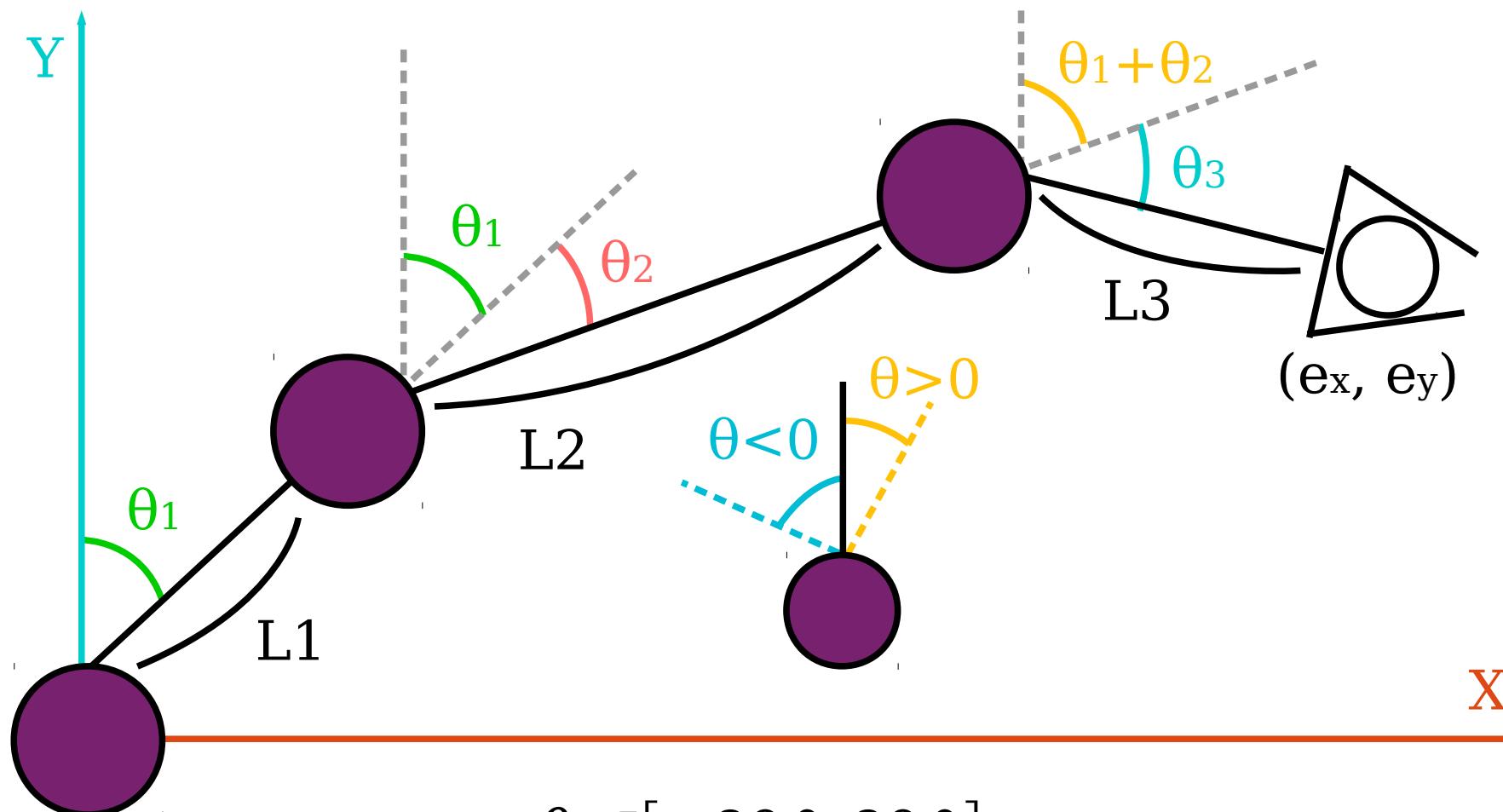
Описывает методы определения пространственного положения хвата манипулятора в зависимости от известного состояния суставов, если длины соединений известны.

$$\theta \in \mathbb{R}^3; \quad e \in \mathbb{R}^2 \quad (22)$$

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}; \quad e = \begin{bmatrix} e_x \\ e_y \end{bmatrix} \quad (23)$$

$$f(\theta) = e \quad (24)$$

≡ Прямая кинематика



$$\theta_i \in [-90^\circ, 90^\circ] \quad (25)$$

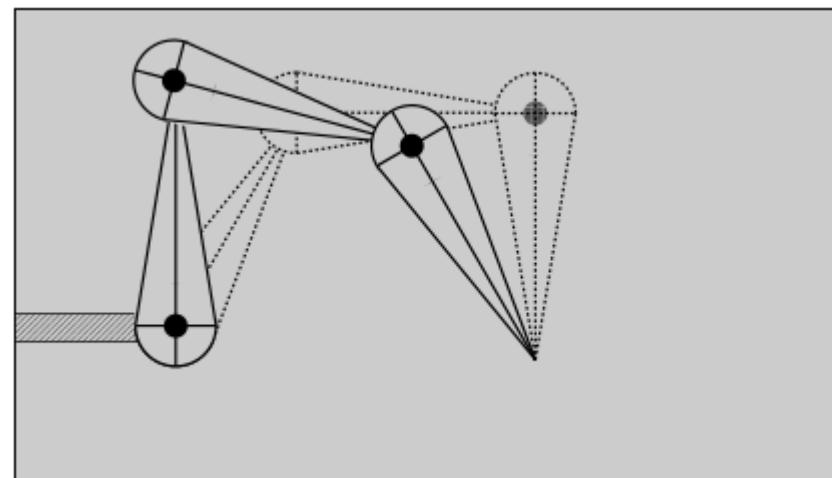
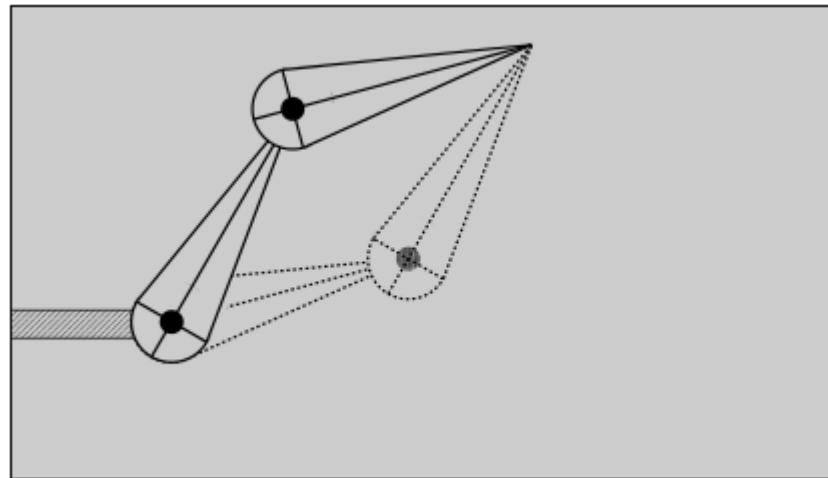
$$e_x = L_1 \sin \theta_1 + L_2 \sin (\theta_1 + \theta_2) + L_3 \sin (\theta_1 + \theta_2 + \theta_3) \quad (26)$$

$$e_y = L_1 \cos \theta_1 + L_2 \cos (\theta_1 + \theta_2) + L_3 \cos (\theta_1 + \theta_2 + \theta_3) \quad (27)$$

≡ Обратная кинематика

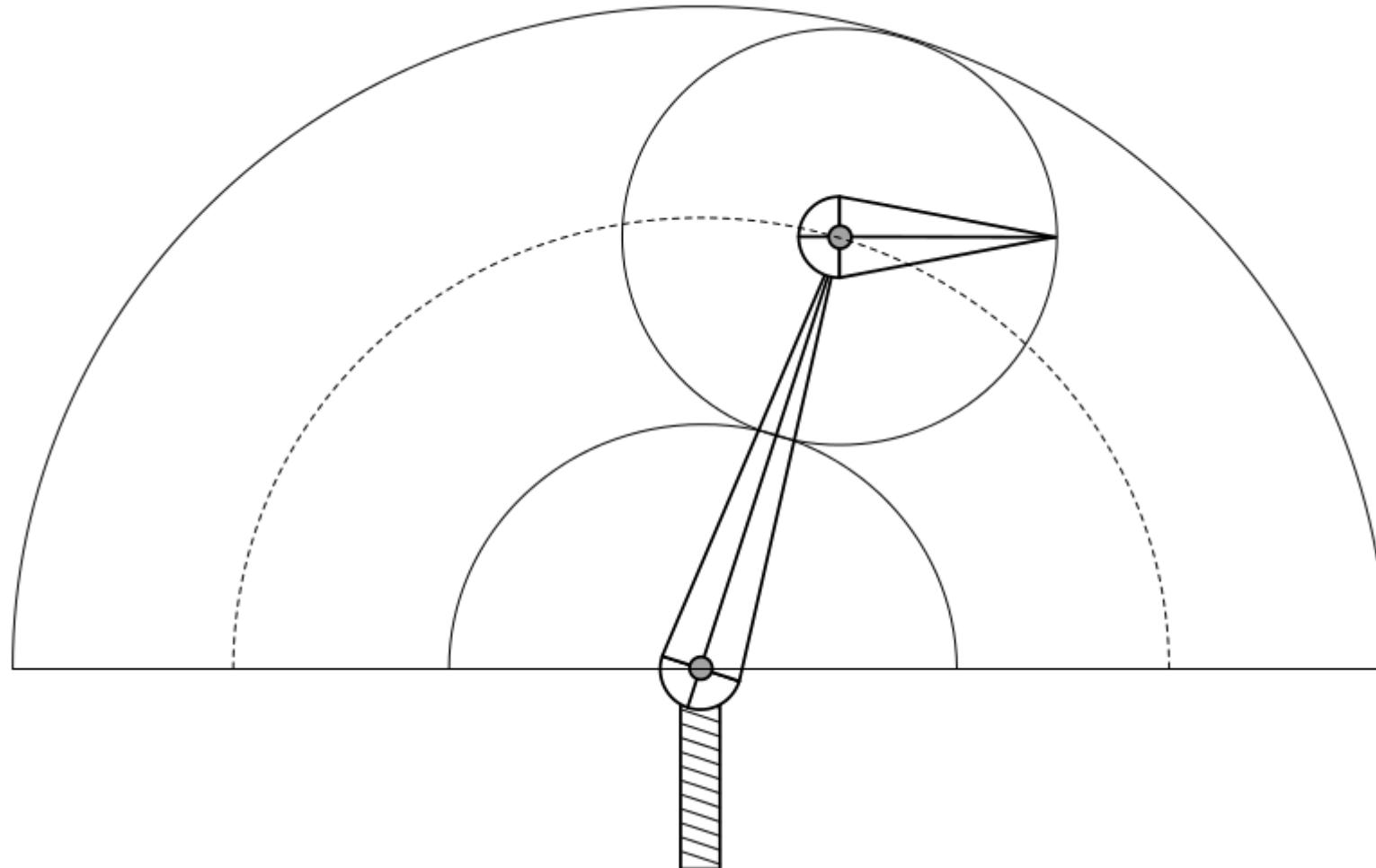
Изучает способы вычисления параметров системы, достаточных для достижения хватом определенного положения в пространстве.

$$\theta = f^{-1}(e) \quad (28)$$



Несколько возможных решений

≡ Обратная кинематика



Возможно отсутствие решений

≡ Метод обратного якобиана

$$f : \mathbb{R}^3 \rightarrow \mathbb{R}^2; \quad e = f(\theta) \in \mathbb{R}^2; \quad \theta \in \mathbb{R}^3 \quad (29)$$

$$J(\theta) = \frac{de}{d\theta} = \begin{bmatrix} \frac{\partial e_x}{\partial \theta_1} & \frac{\partial e_x}{\partial \theta_2} & \frac{\partial e_x}{\partial \theta_3} \\ \frac{\partial e_y}{\partial \theta_1} & \frac{\partial e_y}{\partial \theta_2} & \frac{\partial e_y}{\partial \theta_3} \end{bmatrix} = \begin{bmatrix} \frac{\partial e_x}{\partial \theta_1} & \frac{\partial e_x}{\partial \theta_2} & \frac{\partial e_x}{\partial \theta_3} \\ \frac{\partial e_y}{\partial \theta_1} & \frac{\partial e_y}{\partial \theta_2} & \frac{\partial e_y}{\partial \theta_3} \end{bmatrix} \quad (30)$$

$$de = J(\theta) d\theta \quad (31)$$

$$de = e_{current} - e_{goal} \quad (32)$$

$$\theta_{i+1} = \theta_i + d\theta \quad (33)$$

$$d\theta = J^{-1}(\theta) de \quad (34)$$

$$\theta_{i+1} = \theta_i + J^{-1}(\theta) de \quad (35)$$

≡ Метод обратного якобиана

$$de = J d\theta \quad (36)$$

$$J^T de = J^T J d\theta \quad (37)$$

$$(J^T J)^{-1} J^T de = (J^T J)^{-1} (J^T J) d\theta \quad (38)$$

$$(J^T J)^{-1} J^T de = d\theta \quad (39)$$

$$J^{pi} = (J^T J)^{-1} J^T \quad (40)$$

$$J_{11} = \frac{\partial e_x}{\partial \theta_1} = L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) + L_3 \cos(\theta_1 + \theta_2 + \theta_3)$$

$$J_{12} = \frac{\partial e_x}{\partial \theta_2} = L_2 \cos(\theta_1 + \theta_2) + L_3 \cos(\theta_1 + \theta_2 + \theta_3)$$

$$J_{13} = \frac{\partial e_x}{\partial \theta_3} = L_3 \cos(\theta_1 + \theta_2 + \theta_3)$$

$$J_{21} = \frac{\partial e_y}{\partial \theta_1} = -L_1 \sin \theta_1 - L_2 \sin(\theta_1 + \theta_2) - L_3 \sin(\theta_1 + \theta_2 + \theta_3)$$

$$J_{22} = \frac{\partial e_y}{\partial \theta_2} = -L_2 \sin(\theta_1 + \theta_2) - L_3 \sin(\theta_1 + \theta_2 + \theta_3)$$

$$J_{23} = \frac{\partial e_y}{\partial \theta_3} = -L_3 \sin(\theta_1 + \theta_2 + \theta_3)$$



Спасибо за просмотр