



# Система навигации автономного мобильного робототехнического комплекса

**Работу выполнил:** Миронов С.В., студент кафедры ИУ4  
МГТУ им. Н.Э.Баумана

**Научный руководитель:** Курносенко А.Е., ст. преподаватель кафедры ИУ4  
МГТУ им.Н.Э.Баумана

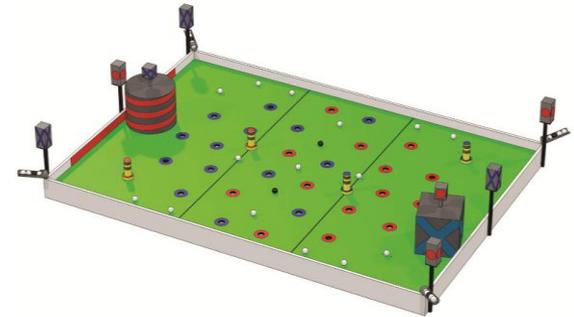
**Рецензент:** Салмина М.А., к.ф.-м.н., с.н.с. Института Механики МГУ

2007 г.



# Предметная область

## Соревнования “Eurobot 2006”:



В ходе одного раунда соревнований, два робота должны набрать максимальное количество очков за 90 секунд. На последних соревнованиях, в задачу робота входили: сбор, сортировка, сброс объектов различного типа: банок, бутылок, батареек, отличающихся друг от друга по форме и цвету.

Размер полигона: \_\_\_\_\_ 300 x 210 мм.

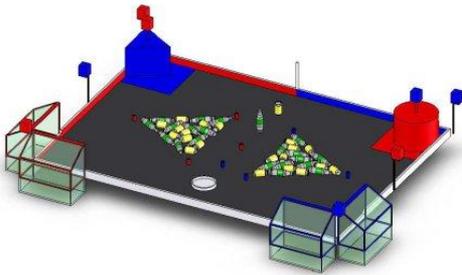
Макс. периметр робота: \_\_\_\_\_ 120 см

Длительность раунда: \_\_\_\_\_ 90 сек

Количество роботов: \_\_\_\_\_ 2

Макс. количество маяков: \_\_\_\_\_ 3

## Соревнования “Eurobot 2007”:





# Задачи системы навигации

## Выполняемые задачи:

- Распознавание объектов окружающей среды;
- Вычисление координат объектов в системе координат, связанной с роботом;
- Определении пройденного пути, текущих координат робота, навигация по карте;
- Распределение приоритетов данных, поступивших из разных источников;
- Формирование и отправка команд модулям нижних уровней
- Отображение положения робота на дисплее, вывод отладочной информации

## Основные разработанные алгоритмы:

- Алгоритм калибровки и компенсации дисторсии камеры
- Алгоритм распознавания и определения локальных координат объектов
- Алгоритм интегрирования данных одометрии
- Система принятия решений, продукционная модель.



# Оборудование - робот «Евронус»

Материнская плата:  
iEi Nano-7270, EPIC  
Intel Celeron 600 МГц  
ОЗУ 512 Мб

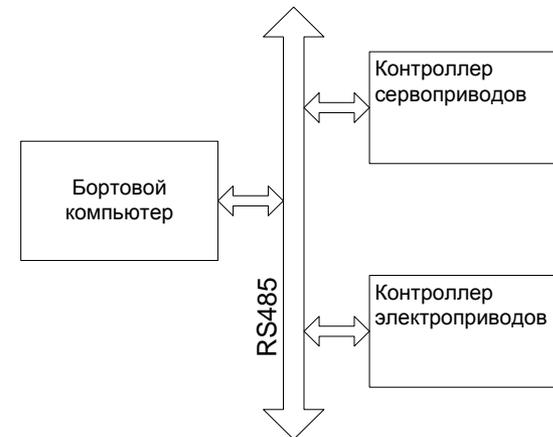
Плата видеозахвата:  
iEi PM-1056, PAL

Сообщение контроллеров и бортового компьютера осуществляется через интерфейс RS485 (модифицированный протокол STX/ETX). Соединение общей шиной по схеме «master-slave».

Напряжение питания: 12В, потребляемый ток: 1,4А

Дальномер:  
Sharp GP2Y0A02YK0F  
дальность: 20-150 см.

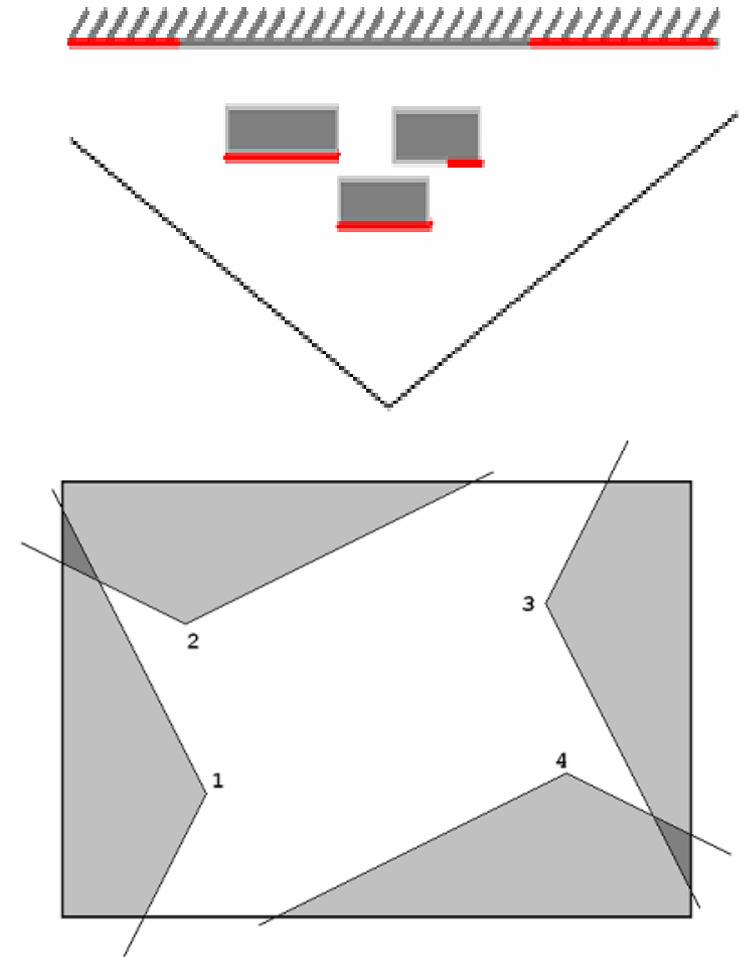
Видеокамера:  
Угол обзора 70  
градусов,  
формат PAL-B





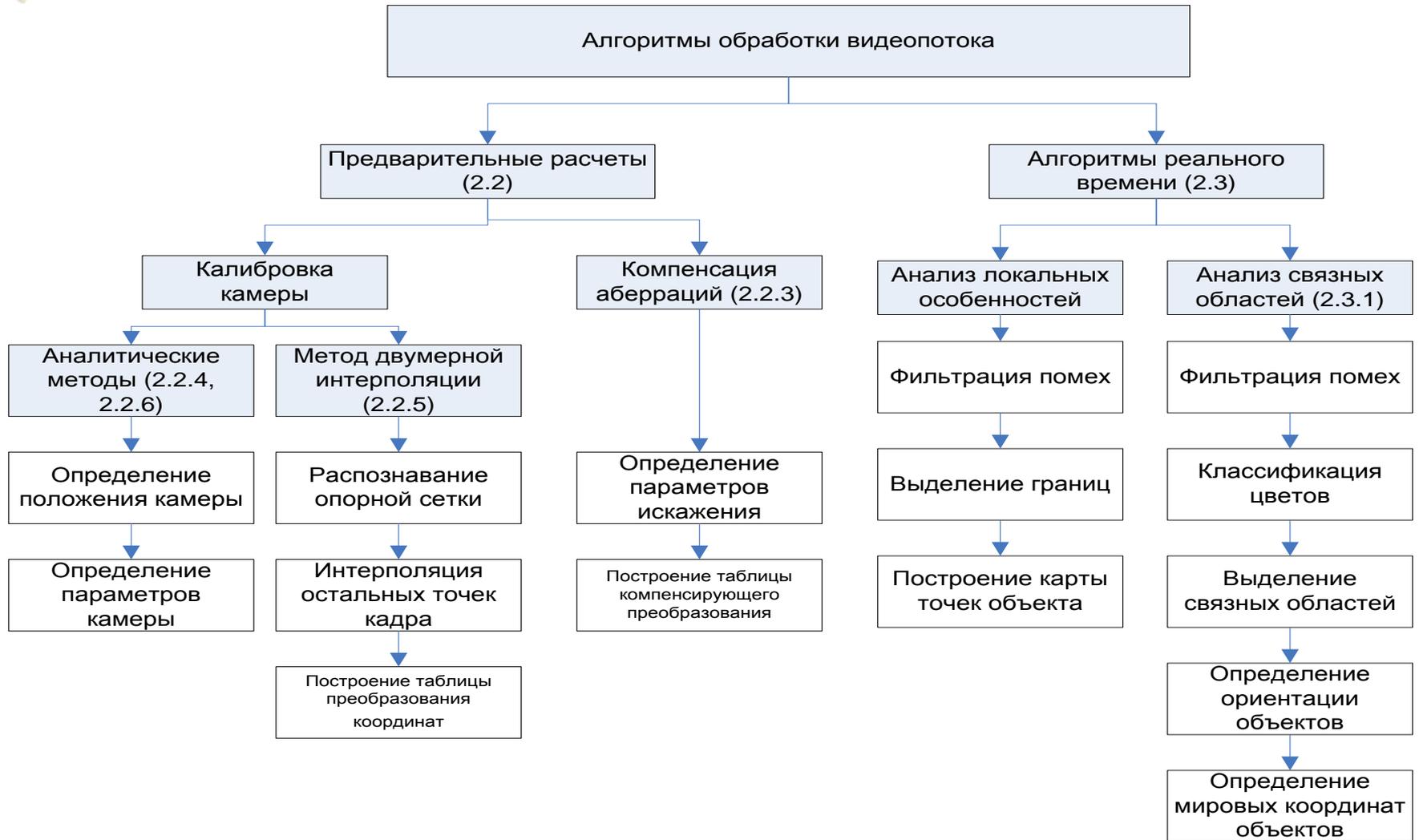
# Дальномер

1. Кластеризация точек: разбиение на группы по принципу близости. Удаление небольших кластеров
2. Линеаризация – приближение каждого кластера точек прямой линией или парой прямых.
3. Группировка кластеров, образующих линию или прямой угол.
4. Поиск группы, лучше всего соответствующей стене или углу полигона





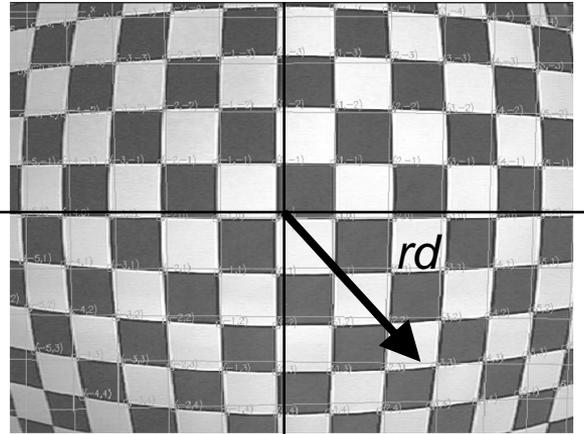
# Классификация алгоритмов обработки видеопотока



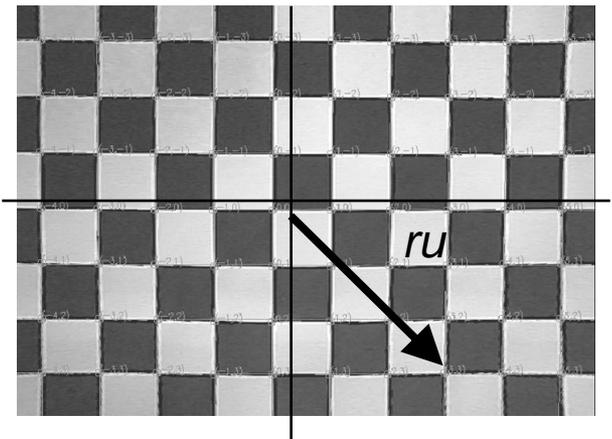


# Компенсация дисторсии

Исходное изображение:



Результат работы алгоритма:



На выходе платы видеозахвата, изображение искажено. Основное искажение – радиальная дисторсия. Так как некоторым алгоритмам необходимо гладкое изображение, выполнялась её компенсация.

Модель бочкообразной радиальной дисторсии:

$$r_u = r_d + \frac{k_3 \cdot r_d^3}{s^2}$$

$k_3$  – к-т дисторсии;  $s$  – масштаб изображения;

Критерий минимизации – показатель отклонения линий от прямых:

$$err = \frac{1}{L} \sum_{j=0}^L \left[ \frac{1}{(N_j - 2)} \sum_{i=0}^{N_j} dist(p_{ij}, line_j)^2 \right]$$

$L$  – количество прямых;  $N$  – количество точек в прямой

до обработки:  $err=5,63$ ; после обработки:  $err=0,69$



# Калибровка камеры. Математическая модель точечной камеры

Калибровка камеры – построение соответствия экранных координат изображений точек мировым координатам оригиналов:

$$z \cdot m = A \cdot [R | t] \cdot M$$

$$z \cdot \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$R = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{pmatrix}$$

$$A = \begin{pmatrix} \frac{f}{s_x} & 0 & c_x \\ 0 & \frac{f}{s_y} & c_y \\ 0 & 0 & 1 \end{pmatrix}$$

$(x, y, z)$  – координаты точки  $M$  в трехмерном пространстве

$(f_x/z, f_y/z)$  – координаты точки  $m$  на изображении

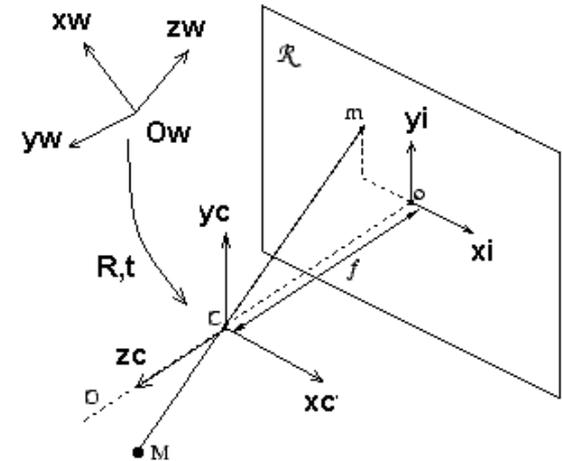
$f_x, f_y$  – фокусное расстояние, в точках, вдоль оси  $x_i$  и  $y_i$

$R$  – матрица поворота глобальной системы координат

$t$  – вектор сдвига глобальной системы координат

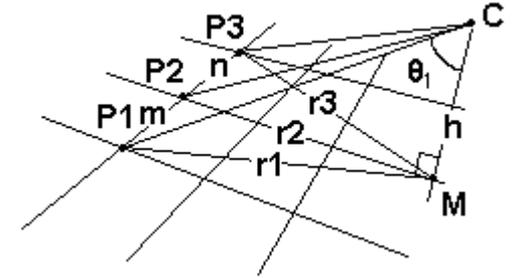
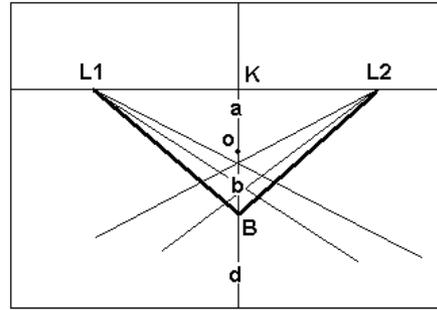
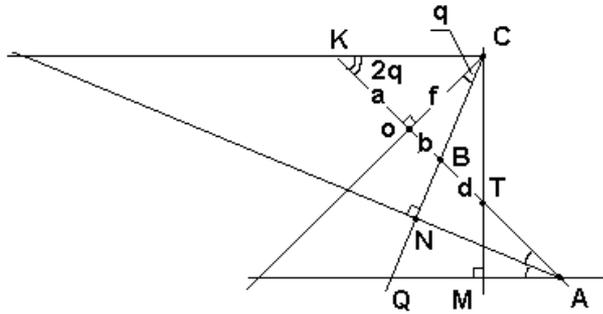
$A$  – матрица перспективного преобразования

$s_x, s_y$  – положение точки - начала координат изображения





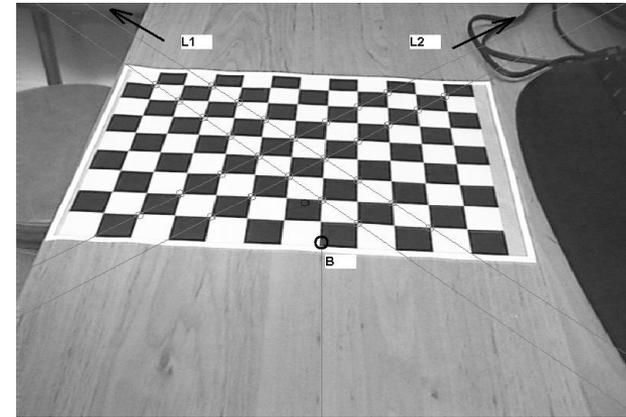
# Калибровка камеры, Аналитический метод.



$$f^2 = 2 \cdot a \cdot b + b^2 \quad \alpha = \frac{\pi}{2} - \arctg\left(\frac{f}{a}\right)$$

$$c_3 = \frac{\operatorname{tg}(\theta_3)}{\operatorname{tg}(\theta_1)} \quad c_2 = \frac{\operatorname{tg}(\theta_2)}{\operatorname{tg}(\theta_1)} \quad r_1^2 = \frac{m^2 \cdot n + m \cdot n^2}{n + m \cdot c_3^2 - n \cdot c_2^2 - c_2^2 \cdot m}$$

Метод позволяет определить параметры камеры (координаты, угол наклона, эффективное фокусное расстояние) по одному изображению шахматной доски без дополнительных измерений. Искажения изображения сильно влияют на точность определения параметров.

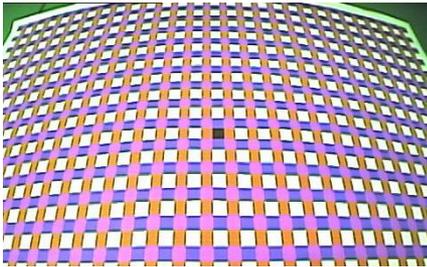


Изображение с нанесенными направлениями к горизонту и точкой пересечения фотоматрицы и плоскости AN (см. рис. выше)

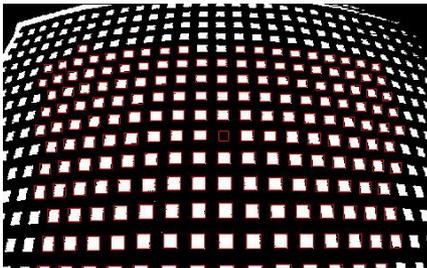


# Калибровка камеры. метод двумерной линейной интерполяции

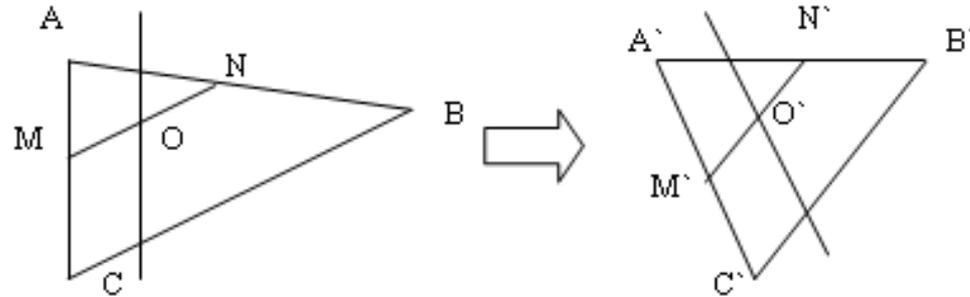
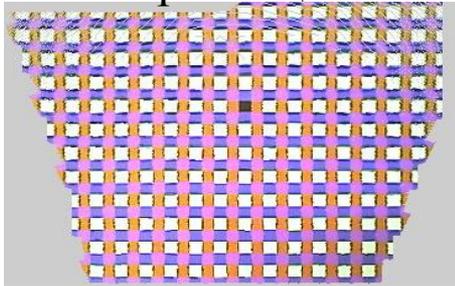
## 1. Исходный кадр



## 2. Выделение точек



## 3. Интерполяция



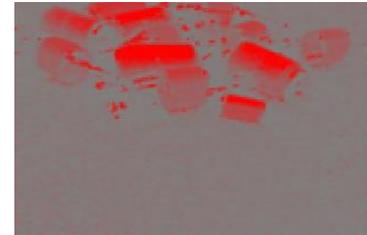
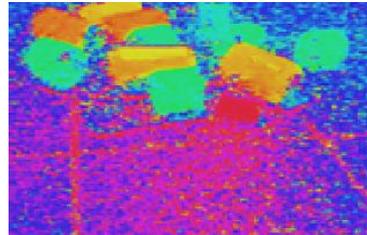
Метод прост в применении и позволяет получить таблицу соответствия координат для большинства точек изображения. Лист бумаги должен полностью находиться в области видимости камеры.

Мировые координаты точки  $O$  определяются из известных мировых и координат  $(A', B', C')$  и экранных координат  $(A, B, C)$  центров белых квадратов и инвариантных пропорций  $AM/MS$  и  $MO/ON$ .



# Распознавание объектов. Цветовое пространство HSL.

Первый этап распознавания объектов – классификация цветов. Для этого удобнее перевести все точки изображения в координаты HSL.



Исходное изображение

Оттенок

Насыщенность

Яркость

$$h = \begin{cases} \text{не определено, если } \min = \max \\ 60 \times \frac{g - b}{\max - \min}, \text{ если } \max = r \text{ и } g \geq b \\ 60 \times \frac{g - b}{\max - \min} + 360, \text{ если } \max = r \text{ и } g < b \\ 60 \times \frac{b - r}{\max - \min} + 120, \text{ если } \max = g \\ 60 \times \frac{r - g}{\max - \min} + 240, \text{ если } \max = b \end{cases}$$

$$s = \begin{cases} 0, \text{ если } l = 0 \text{ или } \max = \min \\ \frac{\max - \min}{2l}, \text{ если } l \leq 0.5 \\ \frac{\max - \min}{2 - 2l}, \text{ если } l > 0.5 \end{cases}$$

$$l = \frac{1}{2}(\max + \min)$$

max и min –наибольшая и наименьшая из компонент RGB соответственно .

Цветовое пространство HSL - Hue, Saturation, Lightness (Оттенок, Насыщенность, Яркость)



# Распознавание объектов. Выделение связных областей, определение ориентации главной оси.

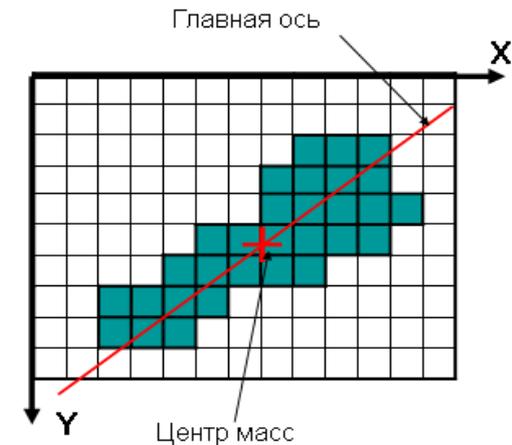
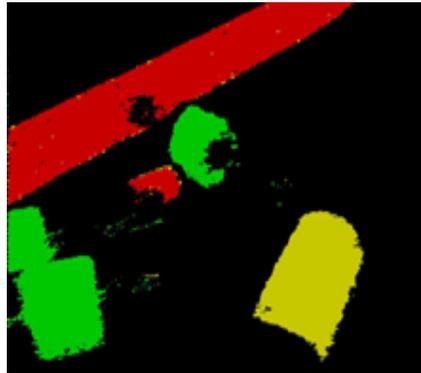
Решение объект/шум принимается исходя из цвета области, её размеров в мировой СК, меры вытянутости.

Моменты инерции:

$$m_{ij} = \sum_{x,y \in S} x^i y^j B(x, y)$$

Угол наклона главной оси:

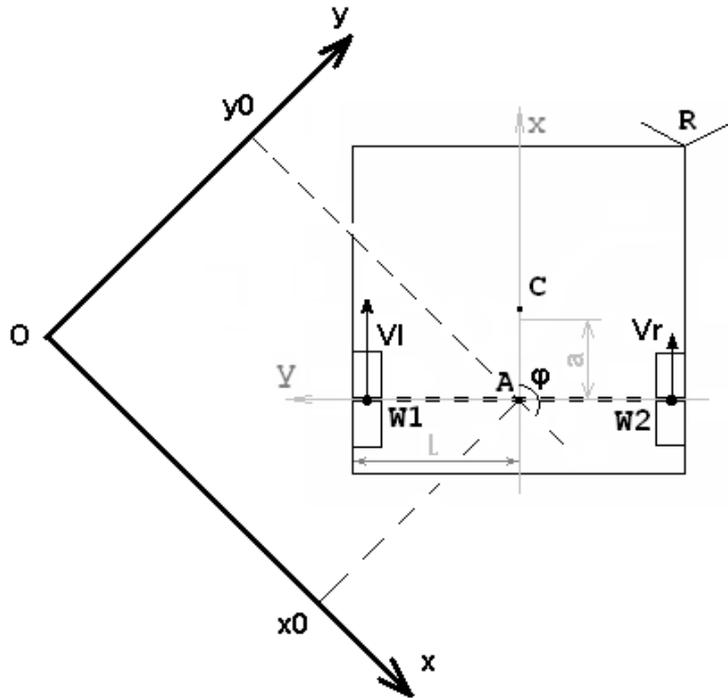
$$\theta = \frac{1}{2} \arctan\left(\frac{2m_{11}}{m_{20} - m_{02}}\right)$$



$m_{i,j}$  – дискретный момент области изображения;  $x, y$  – координаты точки на изображении;  $B(x,y)$  – яркость точки изображения с координатами  $x, y$ ;  $\theta$  – угол наклона главной оси объекта по отношению к оси  $Ox$  изображения



# Навигация по счетчикам импульсов. Схема ходовой части.



На мобильном роботе установлены два датчика, измеряющие угловую скорость вращения ведущих колес. При отсутствии проскальзывания и деформаций колес, можно измерить скорость вращения каждого колеса.

$V_r$   $V_l$  – скорости правого и левого колес,  $l$  – половина расстояния между колесами.  $\varphi$  – угол поворота робота

Точка А – центр локальной системы координат, жестко связанной с роботом

Точка С – геометрический центр робота. Точка О – центр глобальной СК, обычно связанной с точкой старта робота.



## Навигация по счетчикам импульсов. Интегрирование пройденного пути.

Линейная и угловая скорость робота может вычислена по данным счетчиков импульсов на правом и левом колесах

$$V = \frac{1}{2}(V_R + V_L), \quad \Omega = \frac{1}{2l}(V_R - V_L).$$

Кинематические уравнения, связывающие обобщенные скорости:

$$\dot{x} = V \cos \varphi, \quad \dot{y} = V \sin \varphi, \quad \dot{\varphi} = \Omega.$$

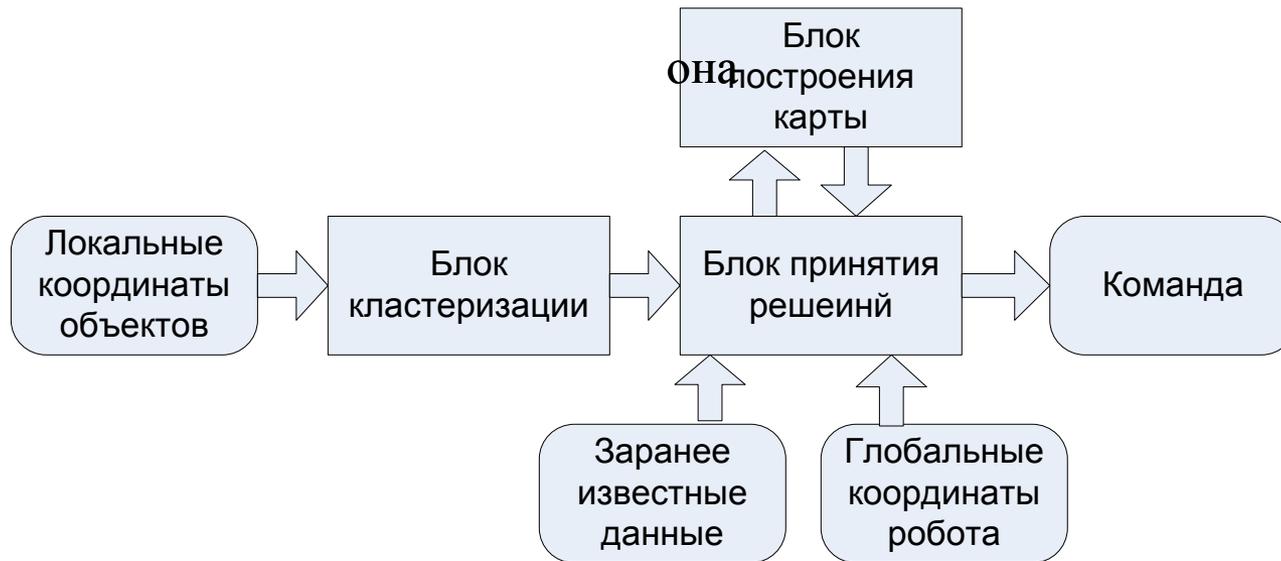
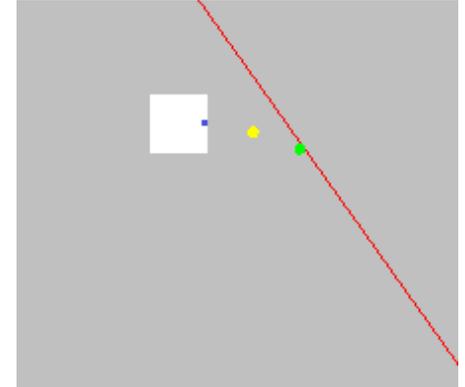
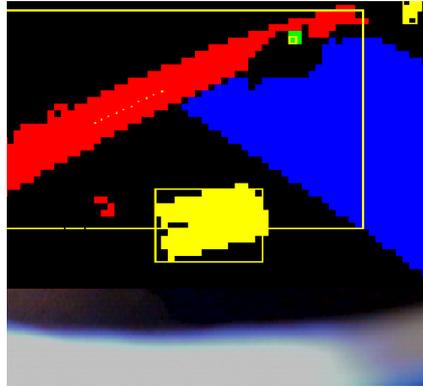
Итоговая система уравнений, интегрируя которые, получим значения координат и угла поворота робота:

$$\dot{x}' = \frac{1}{2}(V_R' + V_L') \cos \varphi', \quad \dot{y}' = \frac{1}{2}(V_R' + V_L') \sin \varphi',$$

$$\dot{\varphi}' = \frac{1}{2l}(V_R' - V_L'),$$



# Построение карты.





# Пример – описание модуля верхнего уровня.

```
bool init(rfloat m)
{
    object<SCHEMA(SSensors)>("Sensors")
        .type(CLSID(Sensors))
        .data<SCHEMA(SSensors)::P_RENDER_MODE>
            (m_render_mode);

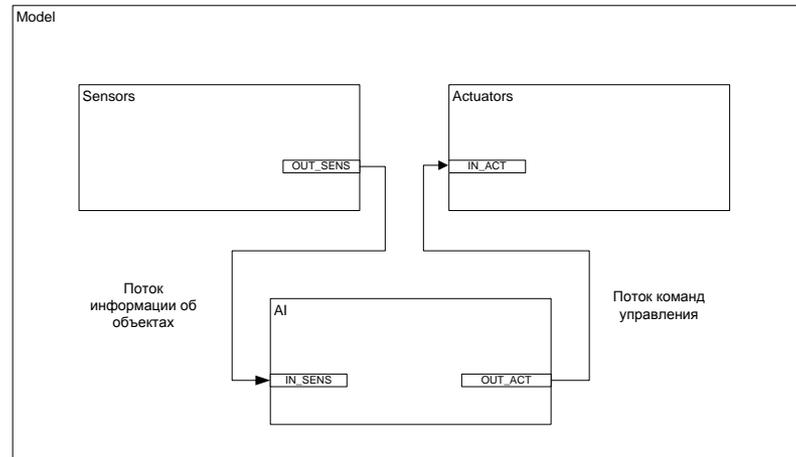
    object<SCHEMA(SAIModel)>("AI")
        .type(CLSID(AI));

    object<SCHEMA(SActuatorsModel)>("Actuators")
        .type(CLSID(Actuators));

    intLink(1) = Link(
        Connector("Sensors",
            SCHEMA(SSensorsModel)::OUT_SENSOR),
        Connector("AI",
            SCHEMA(SAIModel)::IN_SENSOR));

    intLink(2) = Link(
        Connector("AI",
            SCHEMA(SAIModel)::OUT_ACTUATOR),
        Connector("Actuators",
            SCHEMA(SActuatorsModel)::IN_ACTUATOR));

    return true;
}
```





# Результаты

- Реализован набор алгоритмов, позволяющих анализировать данные, поступающие из разных источников информации.
  - Разработаны 3 метода калибровки камеры,
  - Реализован алгоритм компенсации дисторсии объектива
  - Разработана система распознавания объектов, согласно регламенту соревнований «Евробот»
  - Разработана система принятия решений, построенная по продукционной модели.
- Разработана программная модель, основанная на системе вложенных иерархических классов, связывающая компоненты системы. Программа обладает возможностью распределенного запуска.
- Система развернута и опробована на робототехническом комплексе «Евронус», принимавшем участия в соревнованиях «Евробот». Время отклика системы составило меньше 1 сек. Время обработки одного кадра составило 0,017 сек.



# Спасибо за внимание

## **Публикации по теме диссертации:**

В ходе выполнения проекта и сбора материала, были опубликованы следующие работы:

1. Козорезов Ю.Ю., Миронов С.В., Трушкин Ф.А., Юдин А.В. Автономный мобильный робот-гольфист: программно-аппаратный комплекс // Конференция «Мобильные роботы и мехатронные системы 2006». – Сборник научных трудов. Москва, 23-27 октября 2006 г. С. 193 - 202
2. Миронов С.В., Трушкин Ф.А. Система анализа визуальных данных мобильного робототехнического комплекса // Конференция «Мобильные роботы и мехатронные системы 2006». – Сборник научных трудов. Москва, 23-27 октября 2006 г. С. 211-217
3. Миронов С.В. Универсальная система управления робототехническим комплексом // Молодежная научно-техническая конференция учащихся, студентов, аспирантов и молодых ученых "Наукоемкие технологии и интеллектуальные системы 2004". 21-22 апреля 2004 г. С. 117 – 124