

Санкт-Петербургский государственный университет

Математическое обеспечение и администрирование информационных  
систем

Кафедра системного программирования

Хитрук Дмитрий Витальевич

# Цветовая сегментация изображений для робофутбола

Курсовая работа

Научный руководитель:  
д. ф.-м. н., профессор Терехов А. Н.

Консультант:  
главный архитектор ООО "Системы компьютерного зрения" Пименов А. А.

Санкт-Петербург  
2018

# Оглавление

<b>Введение</b>	<b>3</b>
<b>1. Постановка задачи</b>	<b>6</b>
<b>2. Обзор</b>	<b>7</b>
2.1. Существующее решение . . . . .	7
2.2. Влияние цветового пространства . . . . .	7
2.3. Альтернативные решение . . . . .	8
<b>3. Описание решения</b>	<b>9</b>
3.1. Цветовое пространство и модель данных . . . . .	9
3.2. Этап обучения . . . . .	10
3.3. Цветовая сегментация . . . . .	10
3.4. Программная реализация . . . . .	10
3.5. Тестирование . . . . .	11
<b>Заключение</b>	<b>12</b>
<b>Список литературы</b>	<b>13</b>

# Введение

Цветовая сегментация – это процесс разделения цифрового изображения на несколько сегментов в зависимости от преобладающего тона. Цель сегментации заключается в упрощении представления изображения, чтобы облегчить его дальнейший анализ.



Рис. 1: Пример цветовой сегментации.

Цветовая сегментация изображений является важной задачей компьютерного зрения. На данный момент существует множество способов её решения, например, метод Гаусса [1], основанный на распределении цвета, но в условиях интерактивных систем реального времени важными факторами оказываются скорость работы и надёжность используемых алгоритмов.

Робофутбол является системой реального времени. Всё управление роботами производится без участия человека и основано на данных, получаемых с камер, установленных над полем. Для идентификации используются круги различных цветов, размещённые на верхней части роботов. Сами камеры передают информацию на управляющий компьютер с частотой 30 кадров в секунду, из-за чего важной становится эффективность используемых алгоритмов по времени [2].

В робофутболе цветовая сегментация играет большую роль, так как от того, насколько эффективно удаётся распознать объекты на изобра-

жении, зависит правильность дальнейших действий, связанных с управлением роботами и подбором стратегии. Так же в реальных условиях встречаются проблемы, связанные с нестатичностью, неравномерностью освещения и качеством получаемых изображений, поэтому важно учитывать эти особенности системы, чтобы избежать ошибок и критических ситуаций.

Цветовая сегментация в робофутболе используется для того, чтобы различать роботов своих и противника, а также для определения их направления и распознавания мяча, ворот и разметки площадки. На каждом роботе размещены пять цветных кругов, соответственно, задача заключается в отделении этих кругов и распознавании их цветов.

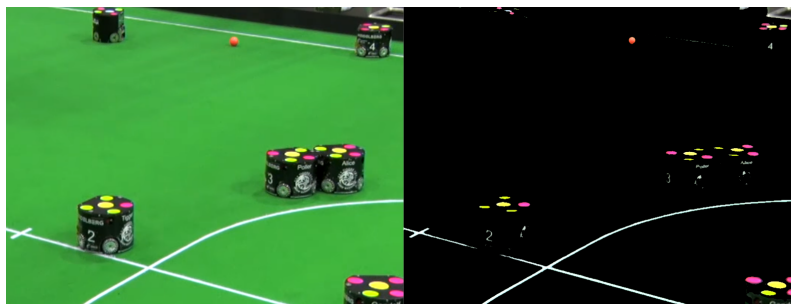


Рис. 2: Результат цветовой сегментации для робофутбола.

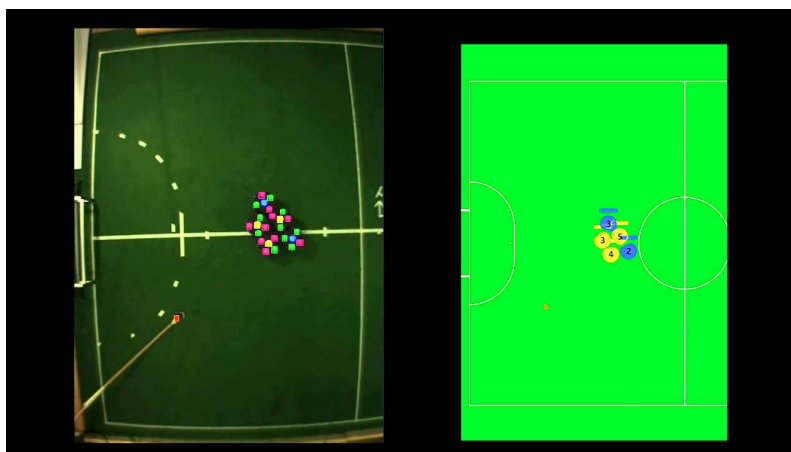


Рис. 3: Обработка полученных данных для дальнейшего использования.

Естественно, что для решения задачи распознавания цветов необхо-

димом сначала вручную установить для каждого участка изображения, какой это цвет. Для этого система сначала обучается с участием человека, который указывает ей, какие цвета представлены на каждом сегменте. Важно максимально корректно собирать информацию на этом этапе, иначе возможны погрешности и ухудшение эффективности по времени.

# 1. Постановка задачи

Целью данной работы является разработка программного инструмента, который позволит распознавать роботов при различных условиях освещённости, при этом необходимо оптимизировать решение для работы в реальном времени.

Для этого необходимо решить следующие задачи:

- Определить, какую информацию необходимо собирать о текущем состоянии системы на стадии обучения для максимальной эффективности и надёжности.
- Разработать алгоритм, способный сегментировать изображение по цветовому признаку на основании предыдущих данных и текущего состояния системы.
- Оптимизировать алгоритм для работы в масштабе реального времени.
- Протестировать решение в реальных условиях, проанализировать.
- Сравнить эффективность и корректность решения с существующими методами.

## 2. Обзор

### 2.1. Существующее решение

Существующее на данный момент решение справляется с распознаванием роботов в условиях, когда влияние внешних факторов сведено до минимума. При изменении освещения, нахождении людей вблизи поля или появлении теней на поле этот подход перестаёт удовлетворительно работать [3].

В большинстве случаев используется двухшаговый алгоритм [4]. На первом шаге происходит процесс обучения, когда вручную задаются цвета для каждого сегмента. Второй шаг - для каждого кадра производится сегментация, которая осуществляется с помощью проверки на попадание отдельных пикселей в заданные диапазоны. Преимущества данного метода:

- Алгоритм прост в реализации.
- Сегментация производится за один проход по изображению, за счёт чего выполняется эффективно по времени.

Недостатком данного подхода является его неустойчивость к изменению освещения, так как исходный цвет при изменении яркости уже не будет попадать в заданные границы. Так же модель RGB приводит к необходимости хранить и проверять шесть параметров, хотя это является избыточным.

### 2.2. Влияние цветового пространства

Выбор цветового пространства - важный этап в задачах машинного зрения. Для цветовой сегментации в условиях нестатичного освещения модель RGB неоптимальна, так как изменения яркости влияют на показатели цветовых компонент, из-за чего данные, полученные на этапе обучения, перестают быть актуальными [5].

Модель HSV лучше подходит для этой задачи, но так как камеры изначально не снимают в данном формате, то необходим перевод из исходного пространства в данную цветовую модель, что является затратной по ресурсам операцией [6].

Для данной работы была выбрана модель YUV, так как перевод из RGB в неё – умножение вектора из компонент на матрицу [5]. К тому же, некоторые камеры изначально снимают в этом формате. Формат YUV менее подвержен негативному воздействию изменения освещения, чем RGB, что так же даёт преимущество в текущей задаче [7].

### **2.3. Альтернативные решение**

Существует решение [8], которое использует похожий метод обхода и сегментации, но, во-первых, оно разработано для состязаний с иными правилами и форматом меток, во-вторых, использует пространство HSV, что замедляет работу алгоритма, в-третьих, на этапе сбора данных используются шесть параметров и метод попадания в границы.



### 3. Описание решения

#### 3.1. Цветовое пространство и модель данных

В качестве цветового пространства было решено использовать YUV, исходя из нескольких причин:

- Многие камеры выдают информацию в данном формате.
- Данное цветовое пространство позволяет легче отделять цвета [5].
- Отдельная яркостная компонента позволяет учесть изменения освещённости.

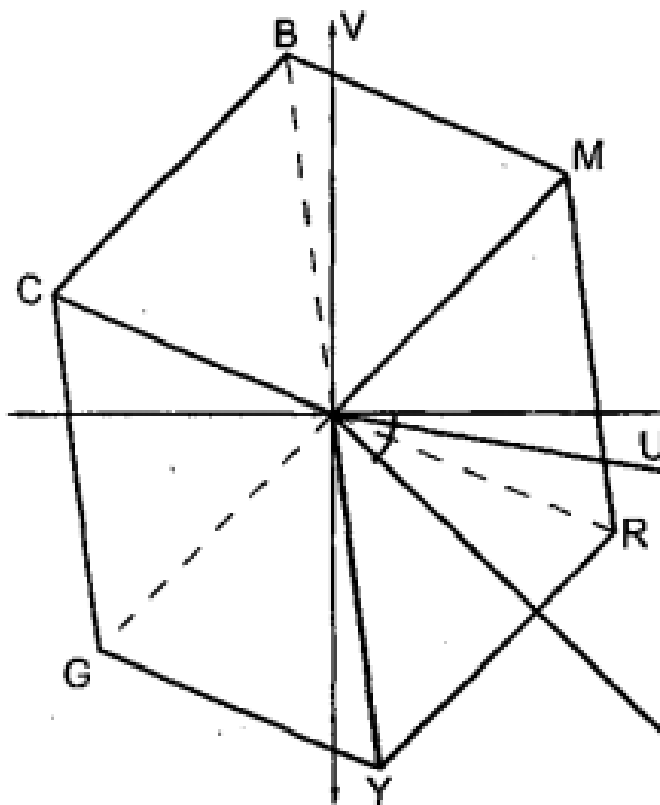


Рис. 4: Сегмент в UV-плоскости.

В ходе работы было выяснено, что для цветовой сегментации достаточно рассматривать только цветоразностные компоненты каждого пиксела. Для корректного определения цвета точки на этапе обучения

определяются диапазоны цветов меток и мяча на UV-плоскости в полярных координатах. Для каждого необходимого цвета хранятся три параметра: начальный угол диапазона, конечный угол, расстояние от начала координат.

## **3.2. Этап обучения**

На этапе обучения собирается информация с камеры, чтобы определить, в каких диапазонах находятся необходимые цвета. Вручную подбираются параметры, описанные в предыдущем пункте, которые обеспечивают оптимальное нахождение на изображении меток и мяча. Таким образом получаем таблицу для цветов меток, мяча и поля, к которой программа обращается непосредственно в процессе игры.

## **3.3. Цветовая сегментация**

Алгоритм сегментации состоит из нескольких шагов. Во-первых, для каждого пиксела мы определяем, отличен ли его цвет от цвета фона. Если это так, то необходимо сначала определить, принадлежит ли цвет точки к некоторому заранее определённом диапазону, чтобы выявить её тип. Далее необходимо обойти соседние точки, чтобы выделить на изображении сегмент и определить его центр. Для этого используются встроенные функции библиотеки `opencv`.

Далее стало понятно, что необязательно просматривать каждый пиксел. Достаточно определить размер мяча – наименьшего объекта, вычислить диагональ его ограничивающего прямоугольника и пропускать по вертикали и горизонтали полученное количество точек.

## **3.4. Программная реализация**

Для реализации данной системы были выбраны язык программирования C++ и библиотека компьютерного зрения `OpenCV` [9].

На этапе обучения мы с помощью встроенных средств для работы с камерой `opencv` получаем изображение и в интерактивном режиме

подбираем параметры для каждого цвета:

1. Вводим параметры.
2. Программа фильтрует изображение и показывает результат.
3. Если результат удовлетворителен, то переходим к следующему цвету, иначе продолжаем подбор параметров.

Во время игры программа получает изображения с камеры и выполняет для них алгоритм, описанный выше. Результат его работы – координаты центра мяча и центров цветных меток. Имея эти данные, можно определить принадлежность роботов к командам и их угол поворота.

### 3.5. Тестирование

При тестировании использовалось следующее оборудование:

- Компьютер с процессором Intel® Core™ i5-2430M CPU @ 2.40GHz × 4.
- Веб-камера с USB подключением и характеристиками 640x480 @ 30 Гц.
- Робот с цветными метками формата RoboCup Small Size League, мяч для настольного тенниса и однотонный пол.

В условиях статичного освещения отслеживание производится корректно, по полученным данным были построены треки передвижения робота и мяча, отклонения не выявлены.

Далее было исследовано влияние изменения освещения на отслеживание объектов. Для этого сначала была смоделирована ситуация, когда поочерёдно изменяется освещение. Были использованы галогеновая, светодиодная и люминисцентная лампы, а так же естественное освещение. Сравнение треков передвижения и реальных траекторий не выявило каких-либо отклонений.

## Заключение

В ходе данной работы были выполнены следующие задачи:

- Определён набор параметров, необходимых для распознавания роботов и сегментации изображения.
- Разработан алгоритм, способный сегментировать изображение по цветовому признаку на основании данных, полученных на этапе обучения, и текущего состояния системы.
- Проведены улучшения, ускорившие работу алгоритма.
- Решение протестировано в реальных условиях.

Представленное решение, хоть и справляется с поставленной задачей, не является законченным программным продуктом. В дальнейшем возможны добавление графического интерфейса и оптимизация полученного алгоритма. Протестировать решение в условиях реального матча не представилось возможным. Планируется использовать программу в реальном матче и сравнить с другими инструментами.

## Список литературы

- [1] Image Segmentation using Gaussian Mixture Models // <https://goo.gl/1q9gQt>.
- [2] Н. Kitano Y. Kuniyoshi I. Noda M. Asada H. Matsubara, Osawa. E. RoboCup: A challenge problem. — 1997.
- [3] Fast and Cheap Color Image Segmentation for Interactive Robots // <https://goo.gl/NKzg9r>.
- [4] GitHub. Ssl-vision Wiki // <https://goo.gl/VWwrgQ>. — URL: <https://goo.gl/VWwrgQ>.
- [5] Thomas P.J. Evolutionary Learning of Fuzzy Control in Robot-Soccer. — URL: <https://goo.gl/f6Enbh>.
- [6] Wikipedia. HSV (цветовая модель). — 2013. — URL: <https://goo.gl/a1AAEs> (online; accessed: 08.02.2018).
- [7] Softpixel. YUV Colorspace. — URL: <https://goo.gl/RnEnBX> (online; accessed: 15.03.2018).
- [8] Белобрюхов М.С., Романенко А.В. Система машинного зрения для киберфутбола. — Доклады ТУСУРа, № 2 (24), часть 2.
- [9] OpenCV. OpenCV library // Open Source Computer Vision Library. — 2018. — URL: <https://opencv.org/> (online; accessed: 08.02.2018).