

Санкт-Петербургский государственный университет

Математическое обеспечение и администрирование информационных систем

Технология программирования

Щавелев Егор Михайлович

Разработка подхода отслеживания
компактных скоростных объектов с
помощью системы камер

Выпускная квалификационная работа

Научный руководитель:
ст. преп. Смирнов М. Н.

Рецензент:
ген. директор ООО "Системы компьютерного зрения" Пименов А. А.

Санкт-Петербург
2017

SAINT-PETERSBURG STATE UNIVERSITY

Software and Administration of Information Systems
Technology of Programming

Shchavelev Egor

Development of high-speed compact objects tracking using cameras system approach

Graduation Thesis

Scientific supervisor:
Senior lecturer Smirnov M. N.

Reviewer:
CEO, Computer Vision Systems LLC Pimenov A. A.

Saint-Petersburg
2017

Оглавление

Введение	5
1. Постановка задачи	7
2. Обзор существующих решений	8
2.1. Физическая интерпретация положения мяча в пространстве	8
2.2. Отслеживание мяча в пространстве	9
2.3. Особенности и отличия предлагаемого подхода	11
3. Основная часть	12
3.1. Физическая интерпретация положения мяча в пространстве	12
3.1.1. Сила тяжести	12
3.1.2. Сила сопротивления воздуха	13
3.1.3. Сила Магнуса	13
3.1.4. Решение системы уравнений	14
3.2. Используемые технологии	14
3.3. Описание используемых алгоритмов и моделей	15
3.3.1. Модель камеры-обскуры	16
3.3.2. Представление изображений объектов на плоскости изображения	18
3.3.3. Внутренние параметры камеры	19
3.3.4. Внешние параметры камеры	19
3.3.5. Триангуляция точек в трехмерном пространстве .	20
3.3.6. Алгоритм скелетонизации	24
3.4. Распознавание объектов в видеопотоке	24
3.4.1. Получение объектов, изменивших свое положение	24
3.4.2. Классификация объектов, изменивших свое положение	25
3.4.3. Определение траектории в двухмерном случае . .	28
3.5. Определение трехмерной траектории мяча с помощью стереопары	29

3.5.1.	Нахождение соответствующих точек на изображениях с двух камер	30
3.5.2.	Предлагаемая реализация подхода построения трехмерной траектории мяча с помощью стереопары .	31
3.5.3.	Проверка правильности работы предлагаемого подхода	32
3.6.	Дальнейшее развитие	33
Заключение		34
Список литературы		35

Введение

В современный мир всё глубже и глубже проникают информационные технологии. Они затрагивают многие деятельности современного человека, начиная от его досуга и заканчивая прикладными программами и алгоритмами, отвечающими за жизнь людей. Поддерживая текущий вектор информатизации и компьютеризации общества и его задач, математики, программисты и многие другие ученые и исследователи занимаются поиском решений и алгоритмов для вычислительных систем, направленных на упрощение и улучшение жизни человека. Так, например, активно развивается интеграция компьютерных систем в сферу соревновательного спорта. Практически в каждом виде спорта сейчас так или иначе присутствуют решения, позволяющие делегировать обязанности судей и людей, ответственных за контроль игровых ситуаций, компьютерной системе.

Для примера рассмотрим такой вид спорта как большой теннис. В 2001 году миру была представлена система Hawk-Eye, позволяющая во время матча разрешать конфликтные ситуации по поводу попадания мяча в корт. Она отслеживает и предсказывает траекторию полета мяча с высокой точностью, и в настоящее время данная система используется во всех официальных теннисных матчах. Однако, данная система, несмотря на ее высокую точность, зрелищность для зрителя и повсеместное применение, отличается своей высокой ценой и большим количеством обслуживающего персонала, необходимого для поддержания работоспособности всей системы.

Этот фактор сильно препятствует ее внедрению в повседневный тренировочный процесс профессиональных игроков, и на текущий момент времени во время тренировки спортсменов на корте есть необходимость использовать людей, в обязанности которых входит:

- Отслеживание активности игроков и их перемещений по корту;
- Отслеживание траекторий полета мячей для оценки эффективности и правильности техники спортсменов;

- Отслеживание физических показателей игроков для оценки их спортивной формы.

Благодаря развитию современных технологий, в частности, компьютерного зрения, часть этих задач можно делегировать компьютерной системе, которая могла бы отслеживать игроков и траектории полета мячей, что позволит освободить человека от такой рутинной и монотонной работы как скрупулезное заполнение многостраничных отчетов о прошедшей тренировке.

Задачу реализации такой системы можно разбить на несколько этапов, первым из которых является разработка системы отслеживания траектории полета мяча в реальном времени. Для реализации такого решения необходимо рассмотреть и реализовать подход к отслеживанию компактных скоростных объектов с помощью системы камер, что и являются целью настоящей работы.

1. Постановка задачи

В рамках настоящей бакалаврской работы ставились следующие задачи:

1. Выполнить поиск и обзор существующих подходов к вычислению траектории движущихся объектов.
2. Разработать алгоритм распознавания объектов в видеопотоке на примере задачи об отслеживании траектории теннисного мяча.
3. Разработать подход к построению трехмерной траектории полета мяча на основе ректифицированной стереопары.

2. Обзор существующих решений

2.1. Физическая интерпретация положения мяча в пространстве

Существующие аналогичные решения представлены в виде научных работ и статей. Например, статья [1] описывает физическую интерпретацию полета мяча при игре в гольф. В частности, модель полета мяча описывается следующим образом:

Мяч представляется в виде сферического упругого объекта, на который во время движения действуют следующие силы:

- F_m – сила Магнуса (подъемная сила);
- F_r – сила сопротивления воздуха;
- F_w – сила тяжести Земли;
- F – равнодействующая всех сил.

По второму закону Ньютона тогда получим следующее уравнение:

$$ma = F_w + F_r + F_m.$$

Данный вывод далее будет использован в настоящей работе в ходе вычисления абсолютного положения мяча в пространстве в физическом подходе решения задачи.

В работе [2] подробно рассматривается физическая модель полета теннисного мяча и приводятся вычисления некоторых коэффициентов и значений для теннисных мячей. Также рассмотрен случай с вращением мяча в двухмерном пространстве, однако вращается он только в одной оси. В физической модели, описанной в настоящей работе, вращение мяча может быть произвольным, что позволит строить траекторию еще точнее.

2.2. Отслеживание мяча в пространстве

В работе Robust 3D Tracking in Tennis Videos[10] рассматривается подход, осуществляющий отслеживание трехмерной траектории мяча в процессе игры в теннис.

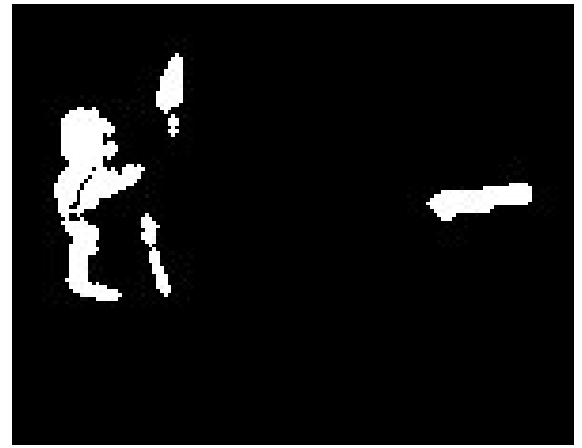
Данный подход основан на предложенном соревновании от компании 3D LIFE, в котором участникам предлагались несинхронизированные видео с 9 камер с процессом игры. Процесс отслеживания траектории описывался следующим образом:

- Выделение движущихся частей из видео;
- Фильтрация этих частей на основе их размеров и цвета;
- Добавление этих частей как кандидатов на мяч;
- Анализ дальнейших кадров для вычисления траектории кандидатов;
- Преобразование траекторий с двух камер в трехмерное пространство и затем проецирование этой траектории на третью камеру. Спроецированные траектории сравниваются с текущими траекториями и траектория с наименьшим отклонением считается действительной траекторией мяча.

Предложенный подход отличается своей эффективностью и простотой, однако у него есть некоторые недостатки, вызванные тем, что данный алгоритм подходит только для данных, предоставленных организатором соревнований. Например, в алгоритме фигурирует некоторое количество констант, касающихся размеров мяча и его цвета. Таким образом, авторы этого алгоритма не смогут обрабатывать некоторые случаи, которые могут возникнуть в ходе эксплуатации этого подхода в различных условиях. Например, сильные удары по мячу или подачи мужчин в условиях освещения в помещении превращают изображение мяча в растянутый закругленный прямоугольник. (Рис. 1).



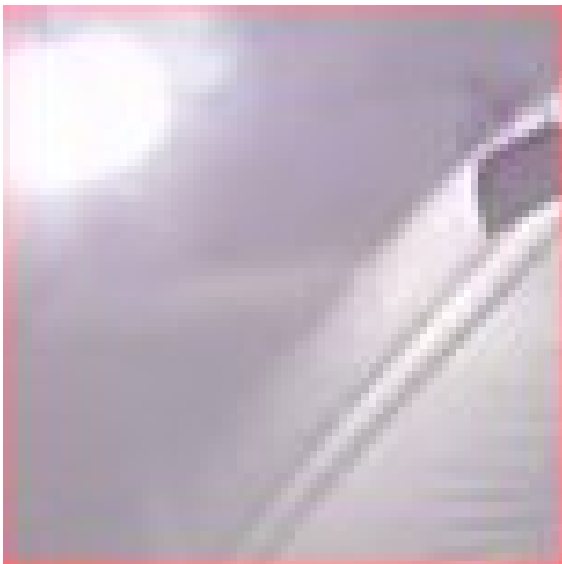
Изображение с камеры



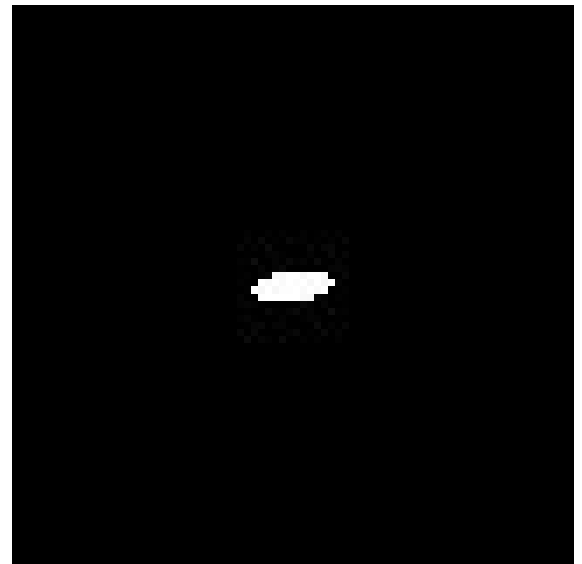
Маска изображения после вычитания фона

Рис. 1

Также возможен случай, что мяч будет пролетать на фоне источника освещения, тогда его изображение почти теряет свой желтый цвет, превращаясь в серое пятно (Рис. 2). Похожая ситуация возникает и при игре на улице, вдали от камеры изображение мяча тоже теряет свои цветовые характеристики.



Пример специфического цвета мяча



Реальное положение мяча после вычитания фона

Рис. 2: Изображение мяча в помещении на фоне источника света

2.3. Особенности и отличия предлагаемого подхода

В рамках настоящей работы предлагается реализовать подход, основанный на более универсальном, по сравнению с аналогами, способе распознавания мяча и более оптимальном алгоритме поиска соответствующих точек для алгоритма триангуляции. После обзора существующих алгоритмов и подходов к определению траектории полета мяча необходимо отметить основные преимущества предлагаемого подхода:

- В нем рассматривается физическая модель полета мяча, готовая к реализации, что позволит увеличить точность определения траектории, а также экстраполировать/интерполировать траекторию для ее дальнейшего анализа;
- Отсутствие привязки алгоритма к константам на моменте детектирования мяча на изображении. Это даст такому подходу дополнительную универсальность и применимость при различных конфигурациях и условиях съемки;
- Более оптимальный поиск соответствующих точек позволит уменьшить время выполнения алгоритма построения трехмерной траектории. Также это позволит также увеличить точность построения траектории в дальнейшем.

3. Основная часть

3.1. Физическая интерпретация положения мяча в пространстве

Отслеживание движущихся объектов с помощью системы камер не представляется возможным без понимания, как эти объекты меняют свое положение в пространстве. Для этого необходимо рассмотреть физическую модель движения таких объектов. В частности, рассматривая модель перемещения мяча в пространстве, в работе [1] уже указаны некоторые формулы и расчеты сил, действующих на мяч для гольфа. Рассмотрим эту модель и перенесем на модель полета теннисного мяча.

Как уже было описано ранее, согласно второму закону Ньютона имеем следующее выражение:

$$F = F_w + F_r + F_m.$$

Рассмотрим проекции всех сил на 3 оси, примем ось Z как ось, коллинеарную силе тяжести Земли.

$$\begin{aligned}F_x &= F_{r_x} + F_{m_x}, \\F_y &= F_{r_y} + F_{m_y}, \\F_z &= F_{w_z} + F_{r_z} + F_{m_z}.\end{aligned}$$

Распишем каждую силу по ее определению и формуле.

3.1.1. Сила тяжести

Исходя из определения силы тяжести – сила тяжести есть произведение массы тела m на ускорение свободного падения g :

$$F_w = mg.$$

3.1.2. Сила сопротивления воздуха

Основываясь на работе [2], имеем следующее выражение для определения силы сопротивления воздуха:

$$F_r = C_d A d v^2 / 2.$$

Где:

- C_d – коэффициент сопротивления воздуха;
- A – площадь поперечного сечения мяча;
- v – скорость мяча;
- d – плотность среды, в нашем случае – воздуха при нормальных условиях.

3.1.3. Сила Магнуса

В работе [2] указана формула для расчета силы Магнуса:

$$F_m = C_l A d v^2 / 2.$$

Где:

- C_l – коэффициент подъемной силы, вычисляемый по формуле

$$C_l = 1 / [2 + (v / v_{spin})],$$

где v – скорость мяча, а v_{spin} – скорость вращения мяча;

- A – площадь поперечного сечения мяча.

3.1.4. Решение системы уравнений

Зная каким образом и какие силы действуют на мяч в течение его полета, можем расписать получившуюся систему:

$$\begin{aligned}F_x &= C_d A d v_x^2 / 2 + 1 / [2 + (v_x / v_{spin_x})] A d v_x^2 / 2, \\F_y &= C_d A d v_y^2 / 2 + 1 / [2 + (v_y / v_{spin_y})] A d v_y^2 / 2, \\F_z &= C_d A d v_z^2 / 2 + 1 / [2 + (v_z / v_{spin_z})] A d v_z^2 / 2 + m g.\end{aligned}$$

Представим данную систему уравнений как систему дифференциальных уравнений и запишем ее, принимая во внимание второй закон Ньютона. Тогда скорость будет являться первой производной по времени, а ускорение – второй:

$$\begin{aligned}m \ddot{x} &= C_d d \pi R^2 v^2 / 2 + 1 / [2 + (v / w R)] d \pi R^2 \dot{x}^2 / 2, \\m \ddot{y} &= C_d d \pi R^2 v^2 / 2 + 1 / [2 + (v / w R)] d \pi R^2 \dot{y}^2 / 2, \\m \ddot{z} &= C_d d \pi R^2 v^2 / 2 + 1 / [2 + (v / w R)] d \pi R^2 \dot{z}^2 / 2 + m g.\end{aligned}$$

В дальнейшем необходимо решить эту систему уравнений, откуда получим координаты мяча в любой момент времени. Получив координаты, становится возможным как интерполировать так и экстраполировать траекторию мяча и использовать её для улучшения точности отслеживания объекта.

3.2. Используемые технологии

В рамках программной реализации настоящей работы были использованы следующие технологии:

1. Python 3 [9] – высокоуровневый язык программирования общего назначения. В частности, с его помощью осуществляется доступ к функциям библиотеки OpenCV.
2. OpenCV 3.1.0 [6] – библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения с открытым кодом.

3. Matlab R2016b [5] – пакет прикладных программ для решения задач технических вычислений и одноимённый язык программирования, используемый в этом пакете.
4. Keras [4] – высокоуровневое API для нейронных сетей, написанное на Python и способное работать поверх TensorFlow или Theano.
5. Theano [11] – библиотека Python, которая позволяет эффективно определять, оптимизировать и оценивать математические выражения с использованием многомерных массивов.

3.3. Описание используемых алгоритмов и моделей

В рамках настоящей работы для решения задачи исследуются и используются некоторые алгоритмы и методы компьютерного зрения. В частности, для выделения движущихся объектов из видеопотока было выбран подход с использованием алгоритмов вычитания фона. Библиотека OpenCV предоставляет следующие алгоритмы:

- BackgroundSubtractorMOG, BackgroundSubtractorMOG2 – алгоритмы работающие на основе модели смесей гауссовских распределений¹ для сегментации объектов переднего/заднего плана.
- BackgroundSubtractorGMG – алгоритм, сочетающий в себе статистическую оценку фонового изображения и попиксельную байесовскую сегментацию.

Представляется возможным использовать и наивное решение, осуществляющее прямую разность между пикселями соседних кадров.

Также для классификации изображений используется сверточные нейронные сети – специальная архитектура искусственных нейронных сетей. Реализуется такая архитектура с помощью чередования способов связей между слоями сети. Благодаря такому устройству, она относительно устойчива к поворотам и смещениям распознаваемых изображений, что позволяет решать задачу классификации в более общем плане

¹Gaussian mixture model.

и не иметь констант и ограничений в отличие от других методов классификации изображений.

В настоящей работе ставится задача определения трехмерного положения объектов с помощью стереопары. Для решения такой задачи необходимо обозначить следующие понятия:

- Модель камеры-обскуры;
- Представление изображений объектов на плоскости кадра;
- Внутренние параметры камеры;
- Внешние параметры камеры;
- Триангуляция точек в трехмерном пространстве.

Рассмотрим подробнее каждое из этих понятий.

3.3.1. Модель камеры-обскуры

Модель камеры-обскуры описывает математическое соотношение между координатами трехмерной точки и ее проекцией на плоскость изображения идеальной камеры-обскуры, где отверстие камеры описывается как точка, и никакие объективы не используются для фокусировки света[8].

Рассмотрим модель простейшей камеры-обскуры (Рис. 3).

Модель содержит следующие объекты:

- Трехмерная ортогональная система координат с началом в точке O . В этой же точке находится отверстие камеры. Точку O называют оптическим центром. Ось Z уходит в сторону просмотра камеры и называется оптической осью. Плоскость XOY является передней стенкой камеры и называется фокальной плоскостью.
- Плоскость изображения $X'RY'$, на которую проецируются точки из трехмерного пространства через отверстие. Эта плоскость параллельна плоскости XOY и расстояние между ними называют фокальным расстоянием f .

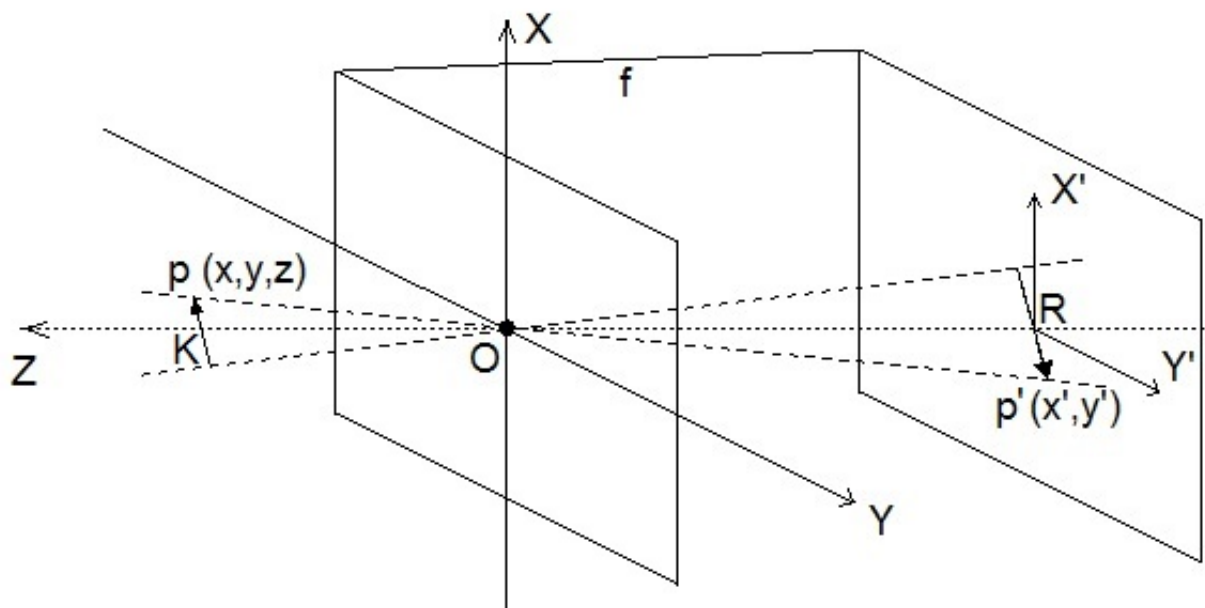


Рис. 3: Модель камеры-обскуры

- Точка R , являющаяся пересечением оптической оси и плоскостью изображения, ее называют центром изображения.
- Точка p с мировыми координатами (x, y, z) .
- Проекционная прямая, проходящая через точки p и O .
- Пересечение проекционной прямой проходящей через точки p и O с плоскостью изображения, а именно проекция точки p на плоскость изображения – точка p' .
- Двухмерная ортогональная система координат с началом в точке R . Оси X' и Y' соответственно параллельны осям X и Y фокальной плоскости. Координаты точки p' в этой системе – (x', y')

Далее, рассматривая данную модель в проекции плоскостей ZOX и ZOY , можно, на основе подобия треугольников, вывести соотношение двухмерных координат точки изображения и ее трехмерного соответствия:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = -\frac{f}{z} \begin{pmatrix} x \\ y \end{pmatrix}.$$

Как видно из рисунка, изображение получается перевернутым. Для того, чтобы упростить некоторые вычисления в дальнейшем, необходимо сделать следующую операцию – плоскость изображения вынести перед фокальной плоскостью (пересекая ось Z в точке f), и тогда формула примет следующий вид:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \frac{f}{z} \begin{pmatrix} x \\ y \end{pmatrix}.$$

Далее будем считать, что эта операция всегда совершается.

3.3.2. Представление изображений объектов на плоскости изображения

Для последующих вычислений положения точки в пространстве предлагается ввести однородные координаты. Теперь, предположим, что мировые точки, и точки на плоскости изображения задаются в однородных координатах. Тогда появляется возможность получить перспективную проекцию двухмерной точки на плоскости изображения в трехмерную в мировых координатах.

Перспективная проекция может быть выражена следующим образом:

$$\begin{pmatrix} fx \\ fy \\ z \end{pmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}.$$

Матрица, описывающая данное преобразование из точек изображения в мировые называется проективной матрицей камеры² P . В этом примере P несет информацию только о фокальном расстоянии камеры. Расширим эти рассуждения, добавив некоторые дополнительные параметры, которые имеют реальные камеры.

²Camera projection matrix.

3.3.3. Внутренние параметры камеры

Перенося вышеописанные рассуждения на модель камеры, приближенной к реальности, необходимо ввести следующие дополнительные параметры:

- Центр изображения (главная точка) – вводится из-за того, что центр изображения не всегда совпадает с началом координатной системы изображения. Характеризуется двумя значениями – c_x и c_y , выражается в пикселях.
- Ширина (s_x) и высота (s_y) одного пикселя матрицы камеры в мировых единицах(мм).
- Угол между осями X и Y , обозначается θ .

Запишем данные параметры в матричном виде:

$$K = \begin{bmatrix} f/s_x & f/s_x \cot(\theta) & c_x \\ 0 & f/s_y & c_y \\ 0 & 0 & 1 \end{bmatrix}.$$

Матрица K называется калибровочной матрицей камеры, она несет в себе информацию о преобразовании координат изображения в координаты пикселей на плоскости изображения. Данное преобразование выполняется при умножении матрицы P слева на K .

3.3.4. Внешние параметры камеры

В реальном мире начало координат камеры может не лежать в начале мировых координат. Поэтому вводятся следующие параметры, выполняющие перенос и поворот координатной системы камеры в соответствии мировой координатной системе:

- Ориентация камеры – описывается матрицей вращения 3×3 R .
- Позиция камеры – описывается трехмерным вектором t .

Запишем данные параметры в матричном виде:

$$G = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}.$$

Матрица G размера 4×4 называется матрицей внешних параметров камеры. Таким образом, проективную матрицу камеры можно записать в следующем виде:

$$P = K[I|0]G = K[R|t].$$

Полное преобразование двухмерных однородных координат в трехмерные для реальной камеры можно записать следующей формулой:

$$m = PM,$$

где m – двухмерные однородные координаты точки на изображении, P проективная матрица камеры, а M – трехмерные однородные координаты.

3.3.5. Триангуляция точек в трехмерном пространстве

Для получения информации о трехмерном положении объекта информации с одной неподвижной камеры недостаточно. Поэтому для решения этой проблемы используется несколько камер. В настоящей работе рассматривается решение задачи с помощью только двух камер, однако в дальнейшем, используя информацию с большего числа камер становится возможным точнее определять траекторию полета мяча. Для определения положения точки на основе изображений с двух камер вводится дополнительная геометрия двух камер (Рис. 4).

Для данной ситуации ставится следующая задача:

- Даны проекции x_1, x_2 точки $X = (X, Y, Z)$ на плоскости изображений камер.
- Необходимо найти трехмерные координаты точки X .

Для решения такой задачи в настоящей работе используется эпиполярная геометрия и понятие существенной матрицы.

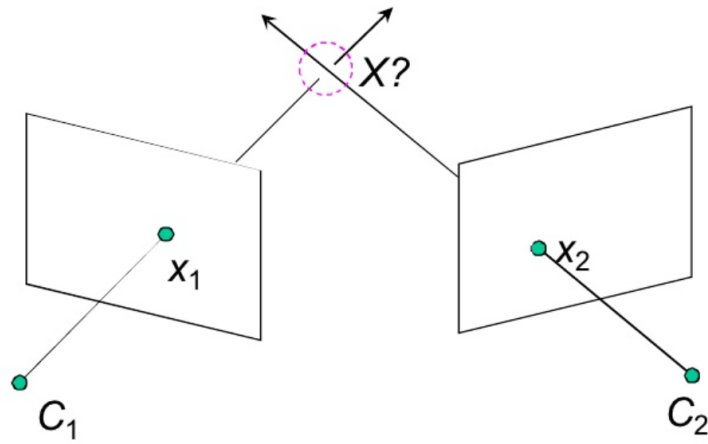


Рис. 4: Задача поиска трехмерного положения точки на основе информации с двух камер

Рассмотрим основные понятия эпиполярной геометрии (Рис. 5).

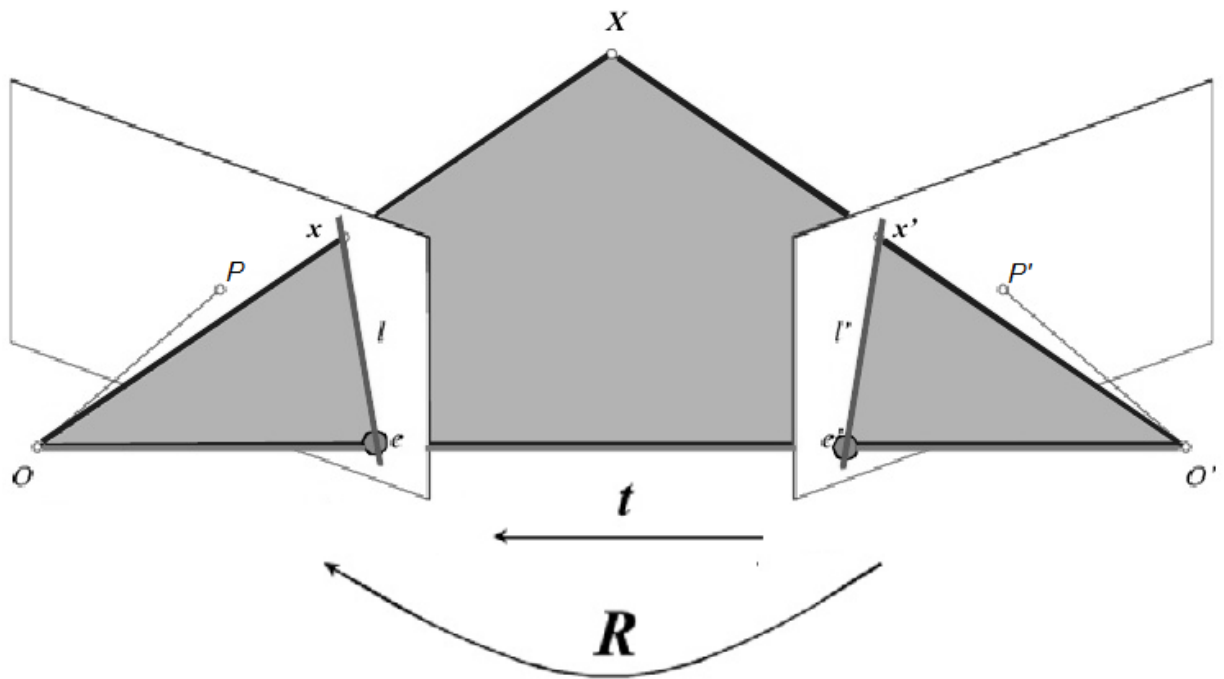


Рис. 5: Эпиполярная геометрия

На данном рисунке:

- Точки O, O' являются центрами двух камер.
- Отрезок OO' называется базовой линией³.

³Baseline.

- Плоскость, описываемая точками $OO'X$ является одной из эпполярных плоскостей (таких плоскостей бесконечно много, все они проходят через базовую линию).
- Проведенные перпендикуляры из точек O, O' в точки P, P' образуют плоскости изображения камер.
- Эпполярные линии l, l' – прямые, образованные пересечением эпполярной плоскости и плоскостями изображений.
- Точки e, e' – точки, образованные пересечением эпполярных линий с базовой линией.

Для такого случая перенесем мировую систему координат в начало координат первой камеры и совместим их.⁴ Умножим матрицу проекции каждой камеры на обратную матрицу внутренних параметров и отобразим точки x, x' на плоскость изображения. После умножения проекционные матрицы первой и второй камеры соответственно примут следующий вид:

$$P_1 = [I|0], P_2 = [R^T | -R^T t],$$

где \bar{t} – вектор сдвига второй камеры относительно первой.

Векторы \bar{x}, \bar{x}' , и \bar{t} находятся в одной плоскости. Преобразуем вектор \bar{x}' из системы координат второй камеры в систему координат первой камеры (т.е. умножим на матрицу поворота R). Исходя из компланарности этих векторов можно сделать следующее утверждение:

$$\bar{x} \cdot [\bar{t} \times (R\bar{x}')] = 0.$$

Перепишем данную формулу в матричном виде:

$$x^T E x' = 0,$$

где $E = [t_x]R$.

⁴Для дальнейших рассуждений необходимо обозначить, что внутренние и внешние параметры камер можно получить, реализовав алгоритм калибровки двух камер с помощью шахматной сетки. Далее считается, что внутренние и внешние параметры камер известны.

Матрица E называется существенной. Существенная матрица дает возможность напрямую получать трехмерное положение точки в пространстве, так как несет в себе информацию о повороте R и направлении сдвига t . Существенная матрица характеризует внешние параметры стереопары и может быть найдена единожды на этапе ее калибровки. Также калибровка стереопары предоставляет длину вектора \bar{t} в мировых единицах β .

Рассмотрим векторы \bar{x} , \bar{x}' и \bar{t} . Преобразуем вектор \bar{x}' аналогичным образом в систему координат первой камеры.

$$\bar{x}'' = R\bar{x}'.$$

Обозначим d и d' за расстояние от точек x и x' до точки X . Далее, исходя из правил сложения векторов, можно сделать следующее утверждение:

$$\frac{\bar{x}}{\|\bar{x}\|}d - \frac{\bar{x}''}{\|\bar{x}''\|}d' = \bar{t}.$$

Отходя от идеальной ситуации, в реальной ситуации прямые, образованные векторами \bar{x} и \bar{x}'' могут быть скрещивающимися. Чтобы внести это условие в вышеописанное утверждение добавим вектор ошибки:

$$\frac{\bar{x}}{\|\bar{x}\|}d - \frac{\bar{x}''}{\|\bar{x}''\|}d' = \beta\bar{t} + \frac{\bar{x} \times \bar{x}''}{\|\bar{x} \times \bar{x}''\|}\gamma.$$

Фактически, γ здесь несет информацию о величине ошибки. Запишем данное утверждение в матричном виде:

$$\begin{pmatrix} d \\ -d' \\ \gamma \end{pmatrix} = W^{-1}t = \begin{pmatrix} x_x & x_{x''} & x_{x \times x''} \\ y_x & y_{x''} & y_{x \times x''} \\ z_x & z_{x''} & z_{x \times x''} \end{pmatrix}^{-1} \begin{pmatrix} \beta x_t \\ \beta y_t \\ \beta z_t \end{pmatrix}.$$

Таким образом, с помощью вышеописанных рассуждений⁵ становится возможным решить задачу поиска трехмерного положения точки на основе информации с двух камер.

⁵Вышеописанные утверждения имеют смысл только если векторы \bar{x} , \bar{x}'' и $\bar{x} \times \bar{x}''$ ненулевые и матрица W обратима. Матрица будет обратимой, так как векторы \bar{x} , \bar{x}'' и $\bar{x} \times \bar{x}''$ линейно независимы.

3.3.6. Алгоритм скелетонизации

Для решения задачи определения трехмерного положения объектов в настоящей работе используется алгоритм Гао-Халла для скелетонизации черно-белого изображения[7]. В рамках настоящей работы реализация этого алгоритма была переписана для языка Python. Результат работы алгоритма продемонстрирован на рисунке 6.

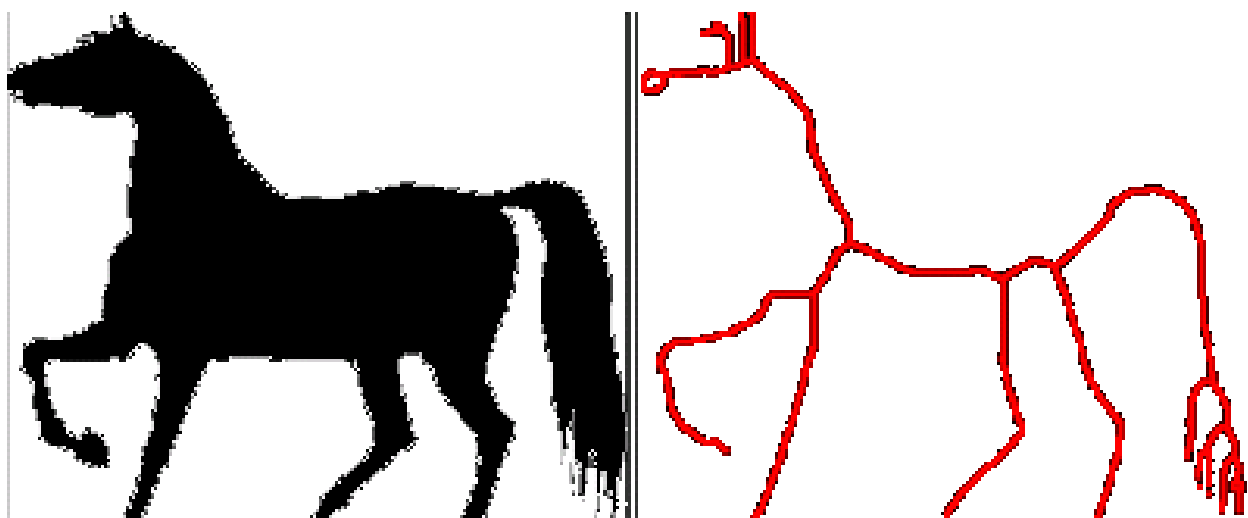


Рис. 6: Результат работы алгоритма скелетонизации

3.4. Распознавание объектов в видеопотоке

Решение этой задачи является одним из шагов алгоритма предлагаемого подхода. Для задачи сегментирования движущихся объектов с помощью неподвижной камеры естественным решением является применение какого-либо алгоритма вычитания фона.

3.4.1. Получение объектов, изменивших свое положение

Имея непрерывный видеопоток представляется возможным выделить объекты, меняющие свое положение от кадра к кадру. Для этого используется библиотека OpenCV и алгоритм BackgroundSubtractorMOG2. Данный алгоритм более адаптивен к изменчивому освещению, что позволяет более четко сегментировать объекты, изменившие свое положение. В параметрах алгоритма можно указать количество кадров, ко-

торое будет участвовать в обработке и пороговое значение расстояния Махаланобиса между одним пикселем изображения и моделью смесей. В ходе исследования параметров было выяснено, что этот параметр косвенным образом отвечает за то, насколько объект должен быть различным относительно фона, что позволяет более тонко настраивать работу алгоритма для разных условий освещения. В рамках настоящей работы были определены какие параметры оптимальны для правильной работы алгоритма в зависимости от освещения, при котором производилась видеосъемка (Табл. 1).

Таблица 1: Оптимальные параметры алгоритма

Освещение (тип)	Уличное	Спортивный зал	Обычная комната
История (кол-во кадров)	100	500	50
Пороговое значение	50	100	5

3.4.2. Классификация объектов, изменивших свое положение

На этом этапе имеется некоторое количество групп пикселей, которые соответствуют реальным объектам, изменившим свое положение. В дальнейшем для построения траектории полета мяча, необходимо соотнести группу пикселей получившихся при вычитании фона и двухмерное текущее положение мяча на изображении. То есть, на получившейся разности кадров необходимо понять, где находится мяч. Для решения этой задачи существует несколько подходов. В рамках настоящей работы было проанализировано и реализовано несколько подходов и выбран наиболее точный подход. Для начала рассмотрим типичную разность кадров видеопотока (Рис. 7).

На кадре можно наблюдать группу пикселей, представленную в виде закругленного прямоугольника, которая представляет собой несколько слившихся снимков мяча на одном кадре. Этот эффект возникает при съемке в помещении при низкой выдержке кадра, из-за того, что за время, пока матрица камеры принимает поток света, мяч успевает переместиться на достаточное расстояние, чтобы его изображение попало



Рис. 7: Вычитание фона между двумя последовательными кадрами

на матрицу камеры за время одного срабатывания затвора несколько раз. Отсюда получается, что мяч иногда может быть представлен не обязательно в форме идеального шара, но и именно в виде закругленного прямоугольника, и это должен учитывать подход, выделяющий мяч на кадре.

Для начала рассмотрим подход, описанный в работе[10]. Выделение мяча на кадре можно производить с помощью сливания группы находящихся рядом пикселей в описывающий их многоугольник и дальнейшую фильтрацию всех получившихся многоугольников по площади и по усредненному значению их цвета на картинке. Однако, как уже было сказано ранее, такой подход не отличается универсальностью, так как он подразумевает, что площадь многоугольника, соответствующего мячу на картинке никогда не превышает некой заданной наперед константы. Таким образом, данное ограничение не предусматривает возможности того, что мяч будет представлен в вышеописанной форме закругленного прямоугольника, так как зачастую площадь соответствующего многоугольника соотносима с площадью многоугольника соответствующего ракетке игрока. В ходе тестирования такого подхода выяснилось, что при возникновении некоторого движения на фоне корта, такие дви-

жения могут определяться как мяч, хотя они могут быть представлены движениями болельщиков, полетом птиц и прочим. Поэтому данный подход неприменим для решения настоящей задачи, поэтому предлагается рассмотреть другой подход.

Следующий подход подразумевает анализ формы группы пикселей получившихся из разности кадров видео. В ходе исследования были выделены различные формы групп пикселей, соответствующих мячу. Далее были выделены прочие объекты, которые не являлись мячом. Нетрудно заметить, что мяч чаще всего представлен в форме шара или растянутого шара в форме вышеописанного закругленного прямоугольника. Отсюда можно поставить задачу классификации формы группы пикселей на два класса:

1. "Мяч".
2. "Не мяч".

Данная задача решалась с помощью современной технологии машинного обучения, а именно – с помощью сверточных нейронных сетей. Для этого была использована библиотека Keras и подготовлена программа на языке Python, помогающая разметить человеку группу пикселей для последующего обучения. В ходе разметки видео было подготовлен следующий набор примеров классов для обучения (Табл. 2).

Таблица 2: Подготовленный набор данных для обучения

Классы	"Мяч"	"Не мяч"
Участвовало в обучении (шт.)	700	1900
Участвовало в валидации (шт.)	200	400

В ходе обучения сверточной нейронной сети была достигнута точность 97% на обучающем наборе данных и точность 95% на валидационном наборе данных, что является достаточной точностью для классификации объектов в рамках решения настоящей задачи.

Для реализации такого подхода был выбран алгоритм, который можно описать следующим образом:

1. Выделение контуров групп пикселей для текущего снимка разности кадров. Данный процесс был реализован с помощью функции библиотеки OpenCV `findContours`, которая на вход принимает двухцветное черно-белое изображение, в параметрах которой можно указать режим поиска контуров и метод их аппроксимации. Для работы алгоритма были выбраны параметры `CV_RETR_TREE` и `CV_CHAIN_APPROX_SIMPLE` по причине необходимости дальнейшей обработки полученной иерархии контуров.
2. Следующим этапом является слияние близко расположенных контуров в один контур. Для этого вычисляется расстояние между центрами контуров и, если такое расстояние меньше заданной константы то такие контуры объединяются. В ходе исследования было выявлено что расстояние в 50 пикселей является оптимальным и удовлетворяет нормальной работе алгоритма для всех видов освещения.
3. На следующем этапе рассмотренные контуры вписываются в окно размером 80x80 пикселей для передачи в нейросеть. После ответа нейросети информация о том, является ли этот контур мячом или нет сохраняется для дальнейшей работы алгоритма.

3.4.3. Определение траектории в двухмерном случае

На этом этапе имеется набор контуров с информацией является ли такой контур мячом или нет. Для построения двухмерной траектории мяча необходимо найти на следующем снимке разности кадров тот же мяч и соединить их центры. Однако, нужно учитывать, что могут возникать как ложно-положительные так и ложно-негативные срабатывания ответа нейросети. Учитывая это, предлагается следующий алгоритм.

- Начало алгоритма. Предположим, что до этого мяч не присутствовал на снимках разности кадров. Теперь, на текущем снимке, появились сведения, что на снимке присутствует мяч. Запомним

его положение и будем ожидать следующего снимка. Дальше возможны следующие ситуации:

1. На снимке отсутствует мяч. Возвращаемся к началу алгоритма.
 2. На снимке присутствует один мяч. Получили траекторию полета мяча на двух снимках, продолжение работы алгоритма.
 3. На снимке присутствует несколько мячей. Выберем тот мяч, чей центр находится наиболее близко к центру предыдущего положения мяча, не учитывая данные о других мячах переходим к пункту 2.
- Ожидаем следующий снимок разности кадров. При получении снимка рассматриваем вышеописанные ситуации.

Работа данного алгоритма продемонстрирована на рисунке 8.

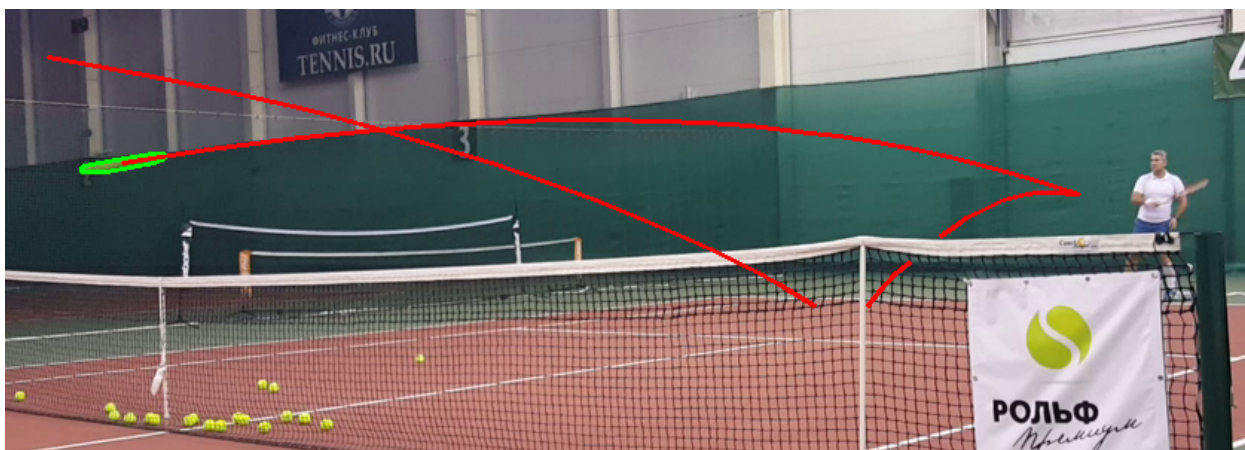


Рис. 8: Траектория мяча

3.5. Определение трехмерной траектории мяча с помощью стереопары

Исходя из описанных рассуждений в главе 3.3 для извлечения информации о трехмерном положении точки в пространстве необходимо иметь проекции одной и той же точки реального пространства на обе

плоскости изображения камер. Задача поиска таких проекций называется задачей нахождения соответствующих точек.

3.5.1. Нахождение соответствующих точек на изображениях с двух камер

Существующие решения поиска соответствующих точек представлены в виде трудоемких задач минимизации функционалов. Также, если необходимо найти соответствия конкретных точек изображения, а в настоящей работе определяется трехмерное положение быстро летящего мяча, представленного на картинке в виде закругленного продолговатого прямоугольника максимальной площадью в несколько десятков пикселей, необходимо решать сложные задачи поиска соотношений и выявлять признаки и характеристики соответствий для их последующего соотношения. Однако, в рамках настоящей работы предлагается упростить задачу поиска соответствий с помощью сокращения набора данных для анализа посредством уменьшения количества точек, передаваемых для решения этой задачи.

Такое преобразование возможно осуществить с помощью алгоритма выделения траектории центра теннисного мяча на основе его изображения в виде закругленного прямоугольника. В рамках настоящей работы было рассмотрено несколько подходов реализации такого алгоритма.

Первый подход предусматривает морфологические преобразования изображения, такие как: эрозия, расширение и их последовательные применения в различном порядке⁶. Однако такой подход отличается тем, что он может привести к потере данных об истинной траектории мяча, например, операция эрозии незначительно уменьшает размеры изображения, а после нескольких последовательных применений этого алгоритма длина истинной траектории мяча может уже сильно различаться с полученной траекторией.

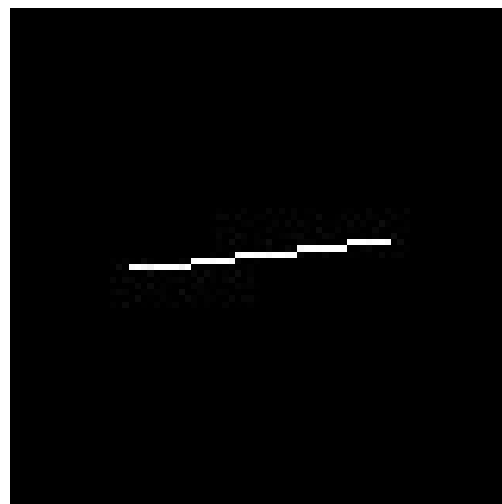
Поэтому предлагается следующий подход: алгоритм скелетонизации Гао-Халла[7]. Результат работы этого алгоритма на изображении мяча

⁶например в библиотеке OpenCV – Opening(эрозия, затем расширение), Closing(расширение, затем эрозия).

можно наблюдать на рисунке 9.



Изображение мяча после вычитания фона



Из изображения выделена траектория его центра

Рис. 9: Результат работы алгоритма скелетонизации

В итоге задача нахождения соответствующих точек на двух снимках теперь сильно упрощается. В настоящей работе данная задача решается следующим способом:

- Для каждого пикселя на картинке с первой камеры проводится эпиллярная линия на изображение второй камеры. Соответствующими пикселями объявляются те, что лежат на одной эпиллярной линии.

3.5.2. Предлагаемая реализация подхода построения трехмерной траектории мяча с помощью стереопары

Таким образом, реализацию предлагаемого подхода предъясняется следующим описанием алгоритма его работы:

1. Перед началом работы алгоритма производится калибровка двух камер с помощью шахматной доски для нахождения существенной матрицы.
2. Начало работы алгоритма. Переход на следующий этап.

3. Поиск мяча на двух камерах. При нахождении мяча на обеих картинках переход на следующий этап.
4. Запуск алгоритма скелетонизации для каждого из найденных изображений мячей. Переход на следующий этап.
5. Вычисление трехмерной траектории мяча. Если необходимо продолжить построение траектории – возврат к этапу 3.

3.5.3. Проверка правильности работы предлагаемого подхода

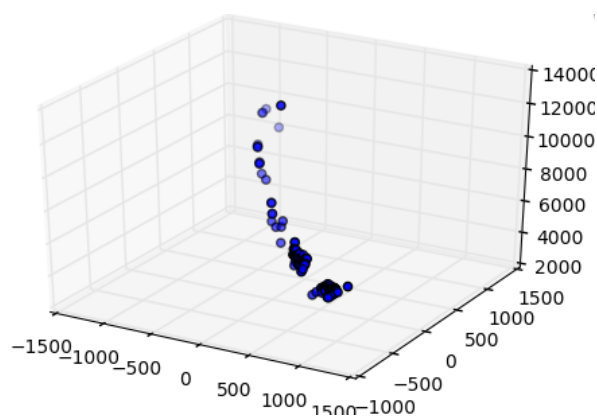
Работа вышеописанного алгоритма была протестирована на основе видеоматериалов, записанных на откалиброванную стереопару (Рис. 10).



Рис. 10: Видеоматериалы с двух камер

Однако, в ходе тестирования не удалось получить доступ к проф. оборудованию и теннисному корту. Так как данный алгоритм предполагает работу во время быстрого полета мяча, то для симуляции такого полета в камере было увеличено время срабатывания ее затвора, что повлияло на то, что мяч в ходе броска рукой растягивался в такие же закругленные прямоугольники как и во время ударов профессиональных теннисистов при нормальных настройках видеокамеры. Построенную траекторию полета мяча можно видеть на рисунке 11. Полученная траектория соотносится с реальным полетом мяча, однако содержит

некоторое количество некорректных данных из-за сильного зашумления картинки вследствие нестандартных настроек камеры, измененных для симуляции.



Трехмерная траектория



Спроецированная траектория

Рис. 11: Результаты работы алгоритма

3.6. Дальнейшее развитие

В дальнейшем планируется увеличить точность работы алгоритма. Увеличение точности можно достигнуть следующим образом:

- Увеличение тренировочной базы для более точного обучения нейросети;
- Использование фильтра Калмана для уменьшения ошибки и предсказания трехмерной траектории;
- Переход на более точный алгоритм триангуляции.

Также планируется протестировать работу алгоритма, записав видео в реальных условиях на теннисном корте с привлечением профессиональных теннисистов.

Заключение

В рамках бакалаврской работы были выполнены все поставленные задачи.

1. Выполнен поиск и обзор существующих подходов к вычислению траектории движущихся объектов. Предложенная физическая модель готова к реализации для уточнения работы алгоритма, вычисляющего трехмерное положение теннисного мяча.
2. Разработан алгоритм распознавания объектов в видеопотоке на примере задачи об отслеживании траектории теннисного мяча.
3. Разработан подход к построению трехмерной траектории полета мяча на основе ректифицированной стереопары.

Список литературы

- [1] Burglund Brett. Golf Ball Flight Dynamics. — 2011. — URL: ["http://home2.fvcc.edu/~dhicketh/DiffEqns/Spring11projects/Brett_Burglund_Ryan_Street/Diff%20Q/pdfscreens/projectoutline.pdf"](http://home2.fvcc.edu/~dhicketh/DiffEqns/Spring11projects/Brett_Burglund_Ryan_Street/Diff%20Q/pdfscreens/projectoutline.pdf) (дата обращения: 28.10.2016).
- [2] Cross Rod. Ball Trajectories. — 2012. — URL: ["http://www.physics.usyd.edu.au/~cross/TRAJECTORIES/42.%20Ball%20Trajectories.pdf"](http://www.physics.usyd.edu.au/~cross/TRAJECTORIES/42.%20Ball%20Trajectories.pdf) (дата обращения: 28.10.2016).
- [3] How to Use Background Subtraction Methods. — 2016. — URL: http://docs.opencv.org/trunk/d1/dc5/tutorial_background_subtraction.html (дата обращения: 28.10.2016).
- [4] Keras. — 2016. — URL: <https://keras.io/> (дата обращения: 20.04.2017).
- [5] Matlab R2016b. — 2016. — URL: <https://www.mathworks.com/> (дата обращения: 12.12.2016).
- [6] OpenCV 3.1.0. — 2016. — URL: <http://opencv.org/> (дата обращения: 12.12.2016).
- [7] Parallel Thinning with TwoSubiteration Algorithms.
- [8] Pinhole camera model. — 2016. — URL: https://en.wikipedia.org/wiki/Pinhole_camera_model (дата обращения: 20.04.2017).
- [9] Python 3. — 2016. — URL: <https://www.python.org/> (дата обращения: 12.12.2016).
- [10] Robust 3D Tracking in Tennis Videos. — 2011. — URL: <https://pdfs.semanticscholar.org/ec6e/dee94b3107678cba420739d139c25afc31e6.pdf> (online; accessed: 20.04.2017).

- [11] Theano. — 2016. — URL: <http://deeplearning.net/software/theano/> (дата обращения: 22.04.2017).
- [12] ball_trajectory.py. — 2016. — URL: https://github.com/gooldan/tennis_track/blob/master/ball_trajectory.py (дата обращения: 12.12.2016).