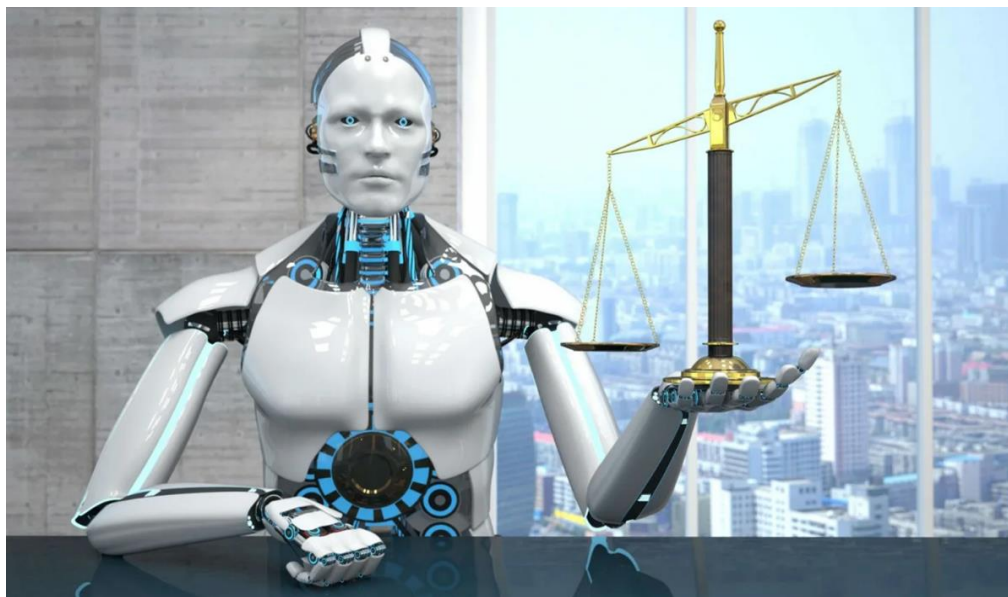


# Создание ИИ-системы по оценке и судейству карате-спаррингов и отдельных приёмов в ката

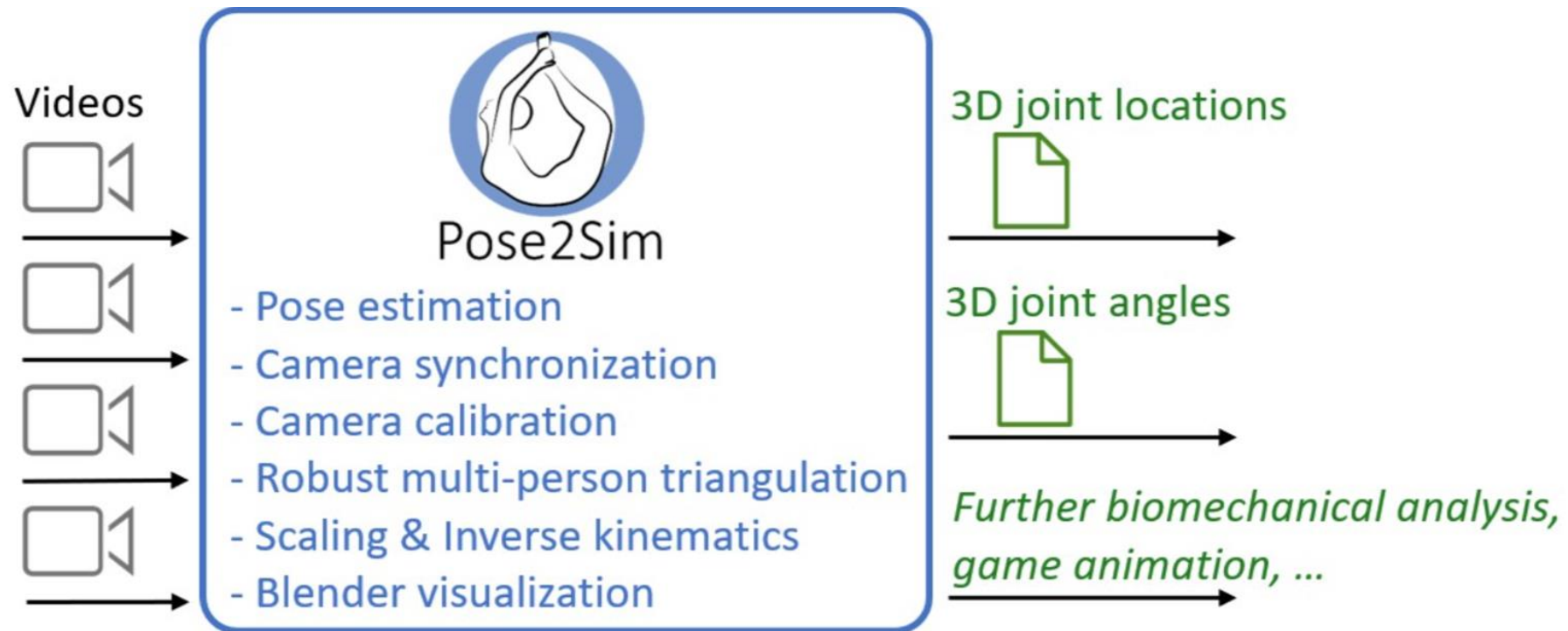


# Цели

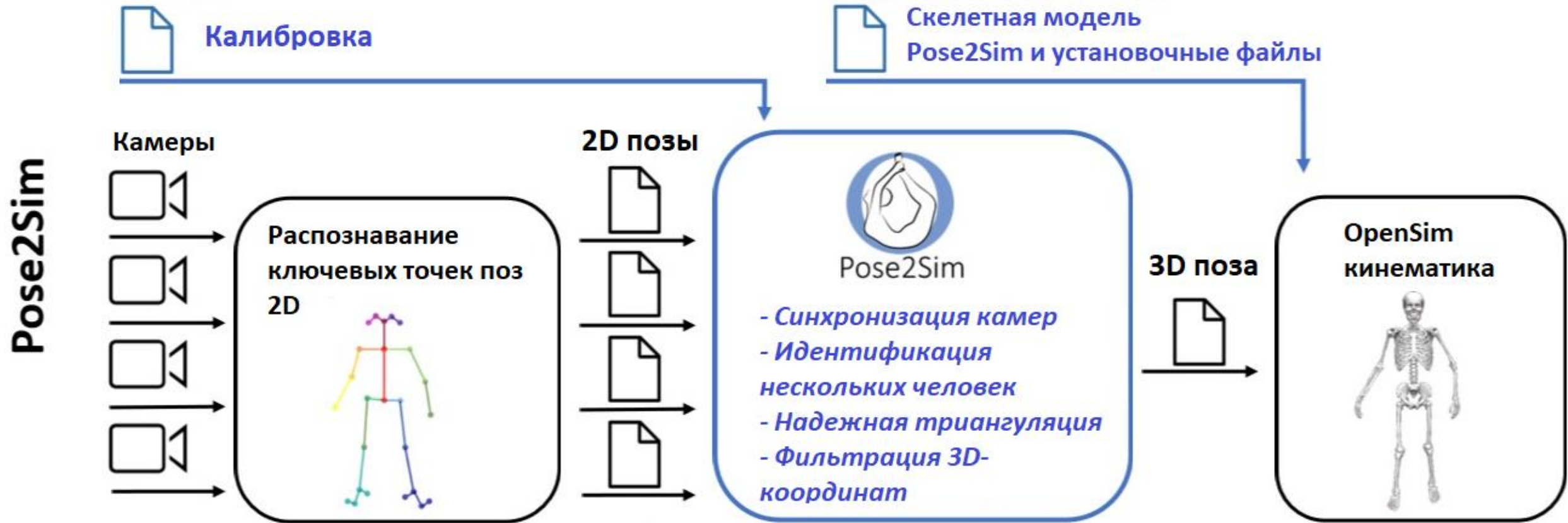
- Разработка системы для автоматической оценки и судейства в карате.
- Использование технологий компьютерного зрения и машинного обучения.
- Анализ видео с различных камер для точной детекции поз и движений спортсменов.

# Pose2Sim

Pose2Sim — это бесплатный и открытый программный конвейер на языке Python для трёхмерного анализа движения человека без использования маркеров.



# Этапы обработки Pose2Sim





# Этапы обработки Pose2Sim

## 1. Калибровка камер

- Внутренняя калибровка
  - Использование шахматной доски или charucoboard
  - Определение внутренних параметров камеры (фокусное расстояние, оптический центр, дисторсия)
- Внешняя калибровка
  - Использование шахматной доски или точек в сцене с известными координатами
  - Определение положения и ориентации камеры в пространстве

## 2. Оценка позы (Pose Estimation):

- Поддержка различных моделей для оценки позы, таких как RTMPose, OpenPose, BlazePose, DeepLabCut и AlphaPose.
- Возможность использования пользовательских моделей, обученных на специфических данных.
- Поддержка многопользовательского анализа.

## 3. Синхронизация видеоданных

- Автоматическая синхронизация
- Вычисление средней вертикальной скорости ключевых точек
- Поиск временного смещения для максимальной корреляции

## 4. Ассоциация персон

- Ассоциация персон через камеры
  - Для многопользовательских сцен
  - Определение минимального расстояния между эпиполярными линиями

## 5. Триангуляция и фильтрация

- Триангуляция
  - Преобразование 2D координат в 3D пространство
  - Использование весов на основе достоверности ключевых точек
- Фильтрация
  - Доступны различные методы фильтрации: Butterworth, Kalman, Gaussian, LOESS, Median

## 6. Интеграция с OpenSim

- Масштабирование модели
  - Автоматическое масштабирование модели по данным триангуляции
- Обратная кинематика
  - Получение угловых данных суставов

# Внутренняя калибровка

Производится с помощью шахматной доски

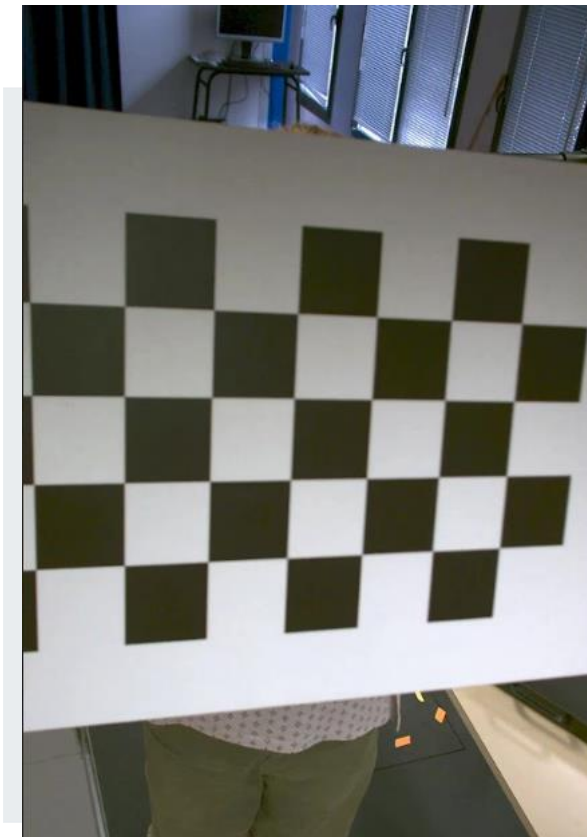
Рекомендации Pose2Sim:

Ошибка внутренней калибровки камер должна быть ниже 0.5 пикселя.

[https://colab.research.google.com/drive/1\\_jctmv4r6t1w8r6AhaRyG-1QcUye3lSt?usp=drive\\_link](https://colab.research.google.com/drive/1_jctmv4r6t1w8r6AhaRyG-1QcUye3lSt?usp=drive_link)

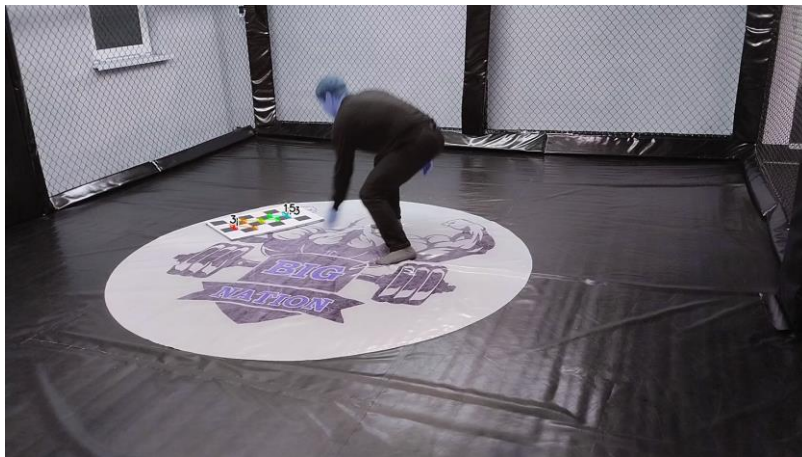
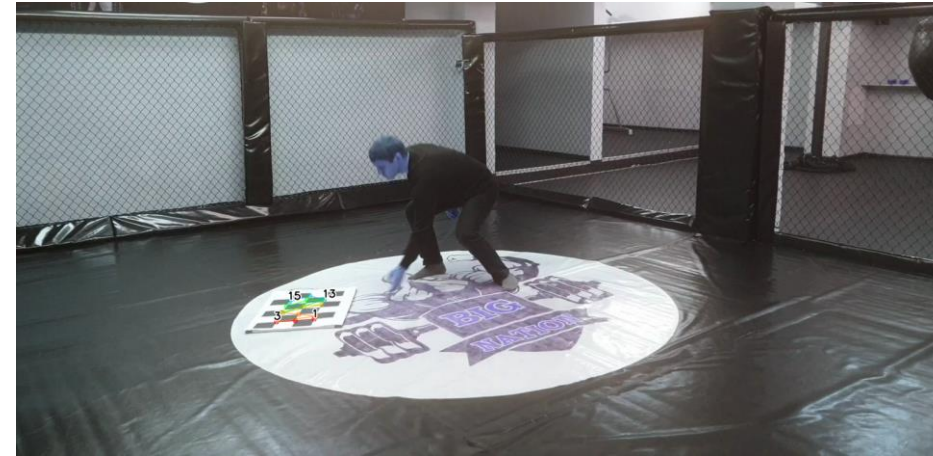


Калибровка в зале:  
ошибка от 1.58 до 2.95 пикселей



Демонстрационная калибровка:  
ошибка от 0.207 до 0.919 пикселя.

# Внешняя калибровка - board

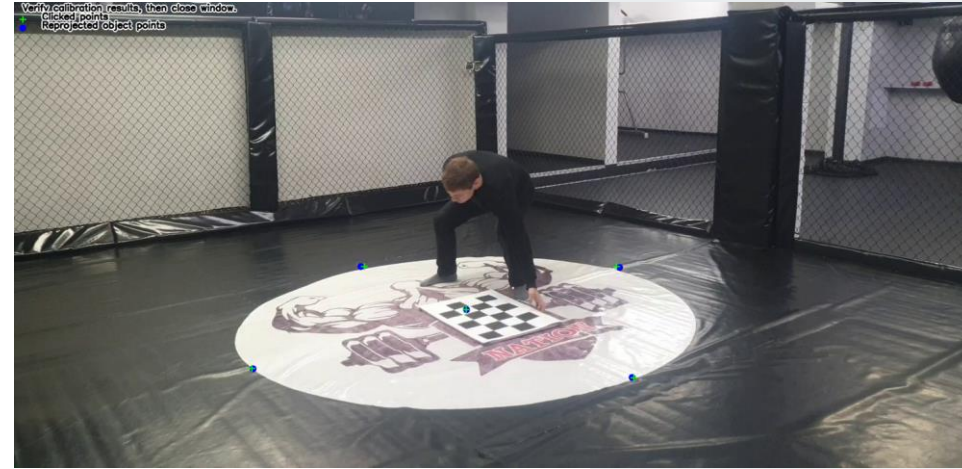


Остаточные  
(среднеквадратичные)  
ошибки калибровки для  
каждой камеры:  
[0.37794962197103316,  
3.056138428859885,  
6.071492806743969,  
6.2815055529481745] px

[https://colab.research.google.com/drive/19\\_jn\\_5K2nndb2UfHHGFrbHeY2W-uMNGY?usp=sharing](https://colab.research.google.com/drive/19_jn_5K2nndb2UfHHGFrbHeY2W-uMNGY?usp=sharing)



# Внешняя калибровка - scene



## Внешние параметры камер:

Для камер было найдено только по 5 опорных точек, что может привести к неточности калибровки. Рекомендуется использовать не менее 10 точек.

Остаточные ошибки калибровки (RMS) для каждой камеры:

ext\_cam01\_img: 12.482 px (28.685 мм)

ext\_cam02\_img: 9.996 px (31.421 мм)

ext\_cam03\_img: 14.319 px (29.296 мм)

ext\_cam04\_img: 11.93 px (23.699 мм)





# Калибровка - результаты

Результаты калибровки сохраняются в файл с расширением toml

**TOML** (Tom's Obvious, Minimal Language) — это формат файлов конфигурации, который предназначен для хранения данных в удобочитаемом виде

Пример:

```
[cam1]
```

```
name = "cam1"
```

```
size = [ 1920.0, 1080.0]
```

```
matrix = [ [ 1537.179793355604, 0.0, 954.5845406116316], [ 0.0, 1537.6812996359145, 458.9784154886433], [ 0.0, 0.0, 1.0]]
```

```
distortions = [ 0.12221887477603854, -0.37444706901953645, -0.02307339292766762, 0.004209686265239807, 0.0]
```

```
rotation = [ -0.8055679964667755, -1.1509501608206665, -1.6685462398400985]
```

```
translation = [ 0.07949527624272877, 0.28340162937055674, 2.432055133213113]
```

Для каждой камеры сохраняются следующие данные:

- name: Имя камеры (например, 'cam1').
- size: Размер изображения камеры в виде списка [width, height].
- matrix: Матрица внутренних параметров камеры (K) в виде списка списков (3x3).
- distortions: Коэффициенты дисторсии (D) в виде списка.
- rotation: Матрица поворота (R) в виде списка списков (3x3).
- translation: Вектор трансляции (T) в виде списка.

# Калибровка - рекомендации

## 1. Размеры шахматной доски

- Количество клеток : Количество клеток на шахматной доске должно быть достаточным для точного определения параметров калибровки. Обычно используются доски размером от 6x9 до 10x7 клеток (например, 6x9 или 8x6). Чем больше клеток, тем более точные данные можно получить.
- Размер клетки : Размер клетки (или расстояние между углами) должен быть выбран таким образом, чтобы обеспечить разрешение изображений, которое позволяет точно обнаруживать углы клеток. Обычно это составляет от 50 мм до 100 мм в зависимости от разрешения камеры и расстояния до объекта.
- При использовании шахматной доски для внешней калибровки необходимо применять доску большего размера, чем при проведении внутренней калибровки

## 2. Планарность доски

- Шахматная доска должна быть идеально плоской. Если поверхность доски будет деформирована или изогнута, это может привести к ошибкам в вычислениях параметров калибровки.
- Убедитесь, что доска не изгибается или не прогибается при перемещении или установке.

## 3. Хорошая контрастность

- Доска должна иметь высокую контрастность между белыми и чёрными клетками. Это поможет алгоритмам обнаружения углов точнее находить их положение.
- Рекомендуется использовать матовые материалы для доски, чтобы избежать бликов, которые могут усложнить обнаружение углов.

## 4. Освещение

- Освещение должно быть равномерным по всей области шахматной доски. Сильные тени или яркие блики могут создавать ложные углы или затруднять обнаружение настоящих углов.
- Избегайте слишком яркого освещения, которое может вызвать пересветы на доске.

## 5. Углы обзора

- Для точной калибровки нужно захватывать шахматную доску под различными углами обзора. Это помогает улучшить оценку внутренних и внешних параметров камеры.
- Захватите доску под углами, которые покрывают как можно большее пространство вокруг камеры (например, с разных сторон, под наклоном, снизу и сверху).

# Калибровка - рекомендации

## 6. Различие позиций

- Позиции доски должны быть разнообразными: доска должна быть расположена в разных частях кадра, под разными углами и на разных расстояниях от камеры.
- Минимум 10-15 различных позиций доски обычно требуется для точной калибровки.

## 7. Отсутствие дефектов на доске

- На доске не должно быть видимых дефектов, таких как царапины, пятна или другие аномалии, которые могут повлиять на точное обнаружение углов.

## 8. Расположение доски в кадре

- Доска должна занимать значительную часть кадра, но не полностью заполнять его. Это позволяет алгоритму точнее определить положение углов.
- Не следует располагать доску слишком близко к краям кадра, чтобы избежать искажений на границах.

## 9. Повторяемость экспериментов

- Если вы выполняете калибровку несколько раз, важно сохранять одни и те же условия съемки и использовать одну и ту же доску. Это позволит сравнивать результаты и проверять их воспроизводимость.

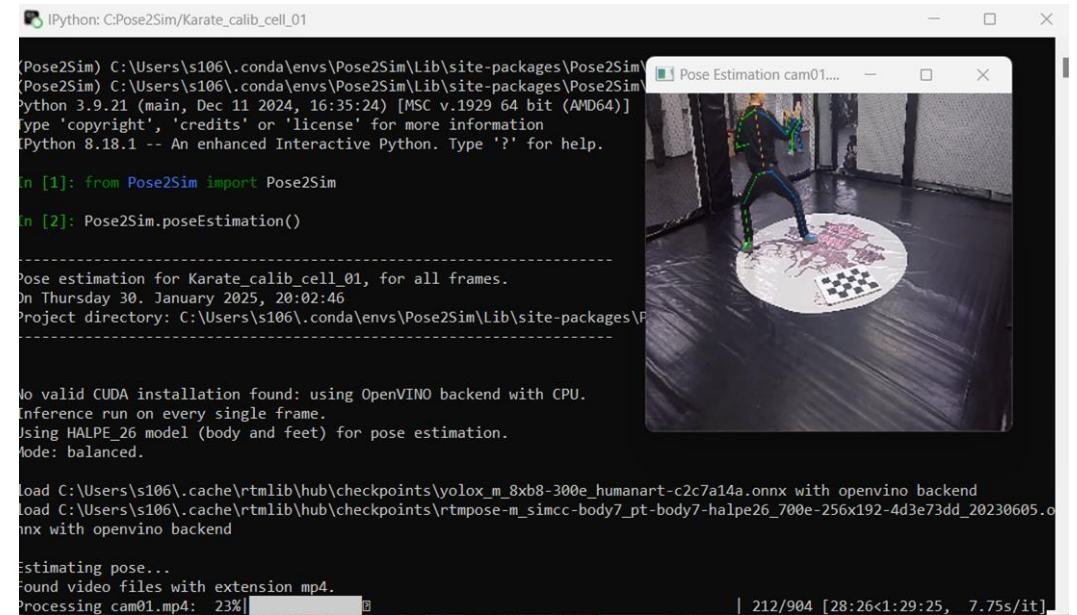
## 10. Использование качественной камеры

- Камера должна иметь хорошее разрешение и минимальное количество аберраций. Низкое качество изображения может значительно увеличить ошибки в определении углов.



# Распознавание ключевых точек поз

Pose2Sim.poseEstimation()



**HALPE\_26** — это модель для оценки позы человека (human pose estimation), которая используется для обнаружения и анализа ключевых точек тела. Она является расширением более известной модели **COCO-Keypoints** и включает в себя 26 ключевых точек, охватывающих не только тело, но также руки и лицо.

## Основные особенности HALPE\_26:

### 1. 26 ключевых точек:

- **Тело:** 17 ключевых точек.
- **Лицо:** 5 ключевых точек (например, глаза, нос, уши).
- **Руки:** 4 ключевых точки (по 2 на каждую руку, например, запястья и локти).

# Синхронизация видео

Pose2Sim.synchronization()

## 1. Анализ ключевых точек:

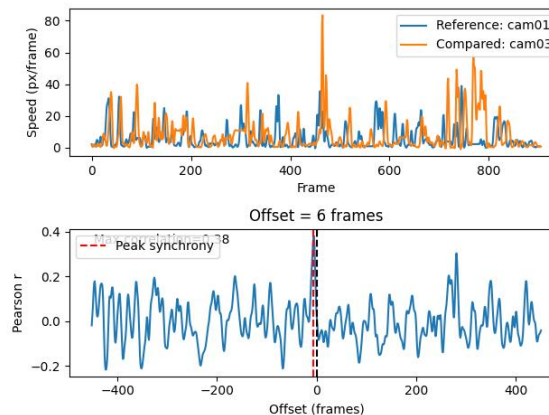
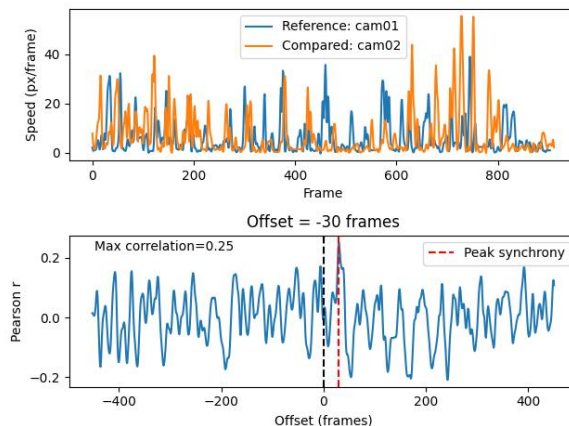
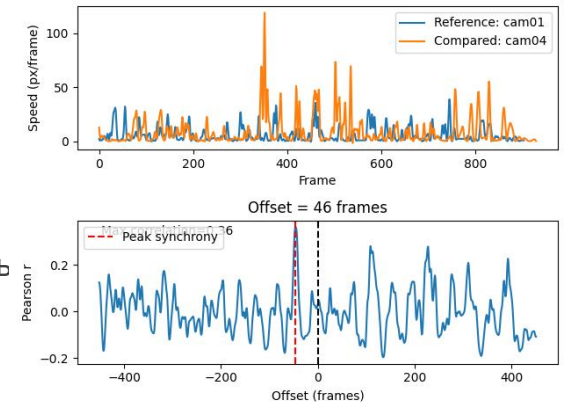
- Фильтрация ключевых точек по порогу уверенности (likelihood), которые были обнаружены с достаточной уверенностью (достоверностью).
- Интерполяция пропущенных значений и сглаживание данных с использованием фильтра Баттерворта.

## 2. Вычисление вертикальной скорости:

- Для каждой камеры вычисляется вертикальная скорость ключевых точек, что помогает определить движение человека.

## 3. Синхронизация камер:

- Сравнение вертикальных скоростей между камерами для нахождения временного смещения (offset), при котором корреляция между данными максимальна.
- Визуализация корреляции и смещения.



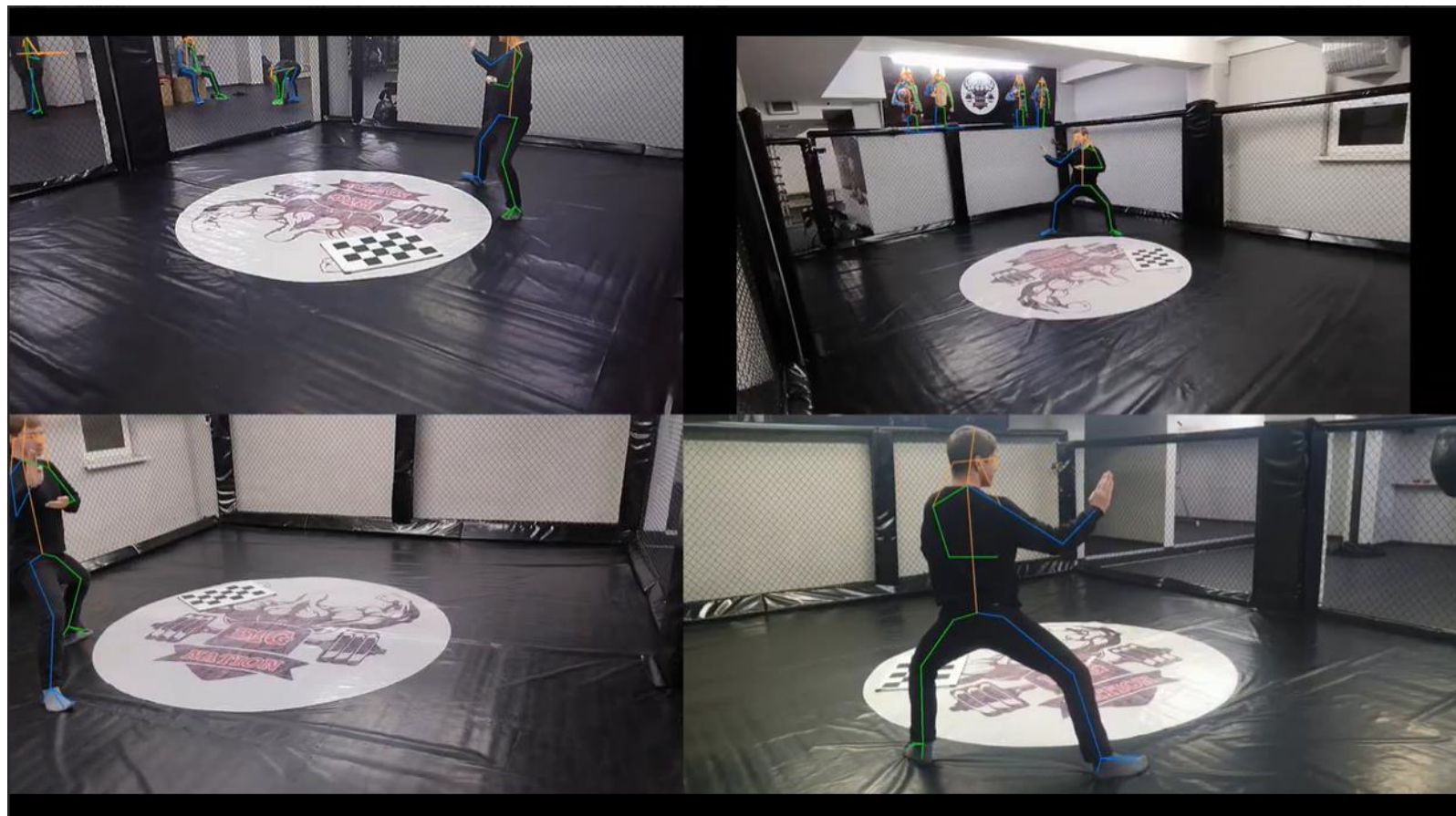
## Результат

--> -> Камеры cam01 и cam02: смещение на -30 кадров, корреляция 0,25.

--> -> Камеры cam01 и cam03: смещение на 6 кадров, корреляция 0,38.

--> -> Камеры cam01 и cam04: смещение кадров на 46, корреляция 0,36.

# Синхронизированное видео



[https://drive.google.com/file/d/1DgFBVu7ZGgv45\\_kzaUzzkxkqyXZHca--/view?usp=sharing](https://drive.google.com/file/d/1DgFBVu7ZGgv45_kzaUzzkxkqyXZHca--/view?usp=sharing)



# Ассоциация персон

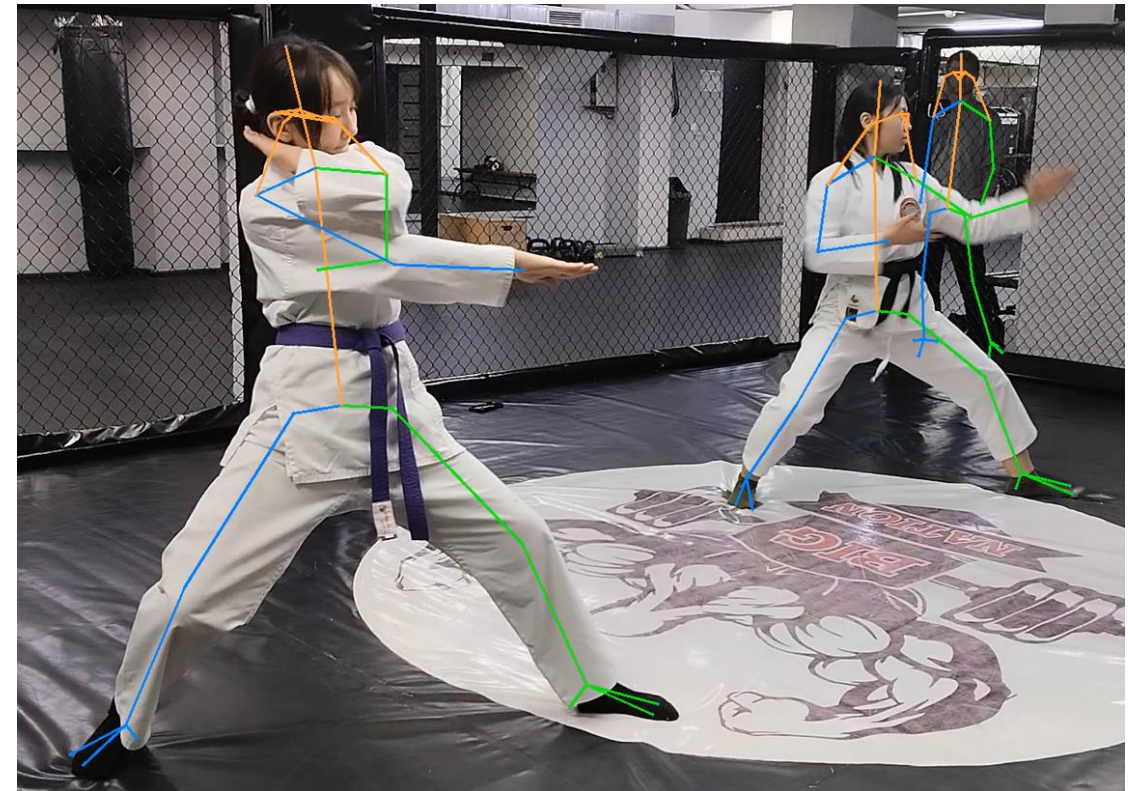
- Ассоциация персон через камеры
  - Для многопользовательских сцен
  - Определение минимального расстояния между эпиполярными линиями

## Результат:

--> Средняя ошибка перепроекции точки шеи на всех кадрах составляет 11,1 пикселя, что примерно соответствует 25,5 мм.

--> В среднем, для достижения требуемого порога погрешности в 20 пикселей пришлось исключить 1,52 камеры после исключения точек с вероятностью менее 0,3.

Pose2Sim.personAssociation()

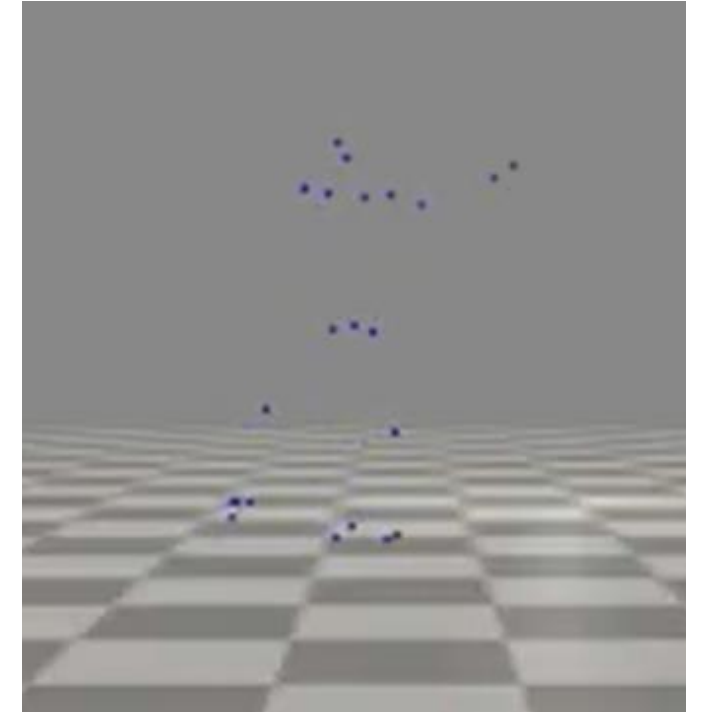


# Триангуляция

Pose2Sim.triangulation()

## Преобразование 2D координат в 3D пространство

- Для каждого кадра и каждой ключевой точки выполняется триангуляция 2D-координат с нескольких камер для получения 3D-координат.
- Используется **взвешенная триангуляция**, где весом является уверенность (likelihood) обнаружения ключевой точки.
- Если ошибка репроекции (разница между исходными 2D-координатами и проекцией 3D-координат обратно на камеры) превышает порог, камеры исключаются из процесса триангуляции до тех пор, пока ошибка не станет допустимой.
- Результаты триангуляции сохраняются в файл формата .trc, который совместим с OpenSim (программное обеспечение для биомеханического анализа).
- Координаты преобразуются из системы координат Z-up (ось Z вверх) в Y-up (ось Y вверх), что является стандартом для OpenSim.



### Результат:

--> Средняя ошибка перепроекции для всех точек на всех кадрах составляет 6,9 пикселей, что примерно соответствует 15,9 мм.

Камеры исключались, если вероятность была ниже 0,3 и если ошибка перепроекции превышала 15 пикселей.

Пропуски интерполировались линейным методом, если они были меньше 10 кадров. Большие пропуски заполнялись последним допустимым значением.

В среднем для достижения этих пороговых значений пришлось исключить 1,93 камеры.

Камера int\_cam01\_img была исключена в 65% случаев, камера int\_cam02\_img - в 57%, камера int\_cam04\_img - в 47% и камера int\_cam03\_img - в 25%.

# Кинематика

```
Pose2Sim.kinematics()
```

**Обработка кинематических данных** с использованием OpenSim, включая масштабирование модели и выполнение обратной кинематики (Inverse Kinematics, IK). Основная цель — получить масштабированную модель человека и рассчитать углы суставов на основе данных из файлов `.trc`, которые содержат 3D-координаты ключевых точек (например, суставов).



## Основные шаги кода:

### 1. Масштабирование модели (Scaling):

- На основе данных из файла `.trc` выполняется масштабирование модели OpenSim под конкретного участника.
- Для каждого сегмента модели (например, рука, нога) вычисляется среднее значение длины на основе кадров, и модель масштабируется в соответствии с этими значениями.
- Результат сохраняется в виде масштабированной модели `.osim`.

### 2. Обратная кинематика (Inverse Kinematics, IK):

- Используя масштабированную модель и данные из файла `.trc`, выполняется обратная кинематика.
- Маркеры модели OpenSim "следуют" за маркерами из файла `.trc`, учитывая кинематические ограничения модели.
- Рассчитываются углы суставов, которые сохраняются в файле `.mot`.

Файл `.mot` содержит данные о **кинематике суставов** (углах и положениях), рассчитанные с помощью обратной кинематики (Inverse Kinematics, IK) в OpenSim.

- `hip_flexion_r`, `hip_adduction_r`, `hip_rotation_r`: Углы сгибания, приведения и вращения правого бедра.
- `knee_angle_r`: Угол сгибания правого колена.
- `ankle_angle_r`: Угол сгибания правого голеностопа.
- Аналогичные параметры для левой стороны тела (`_l` вместо `_r`).
- Другие параметры, такие как углы для позвоночника, шеи, рук и т.д.





# Визуализация с OpenSim



# Выводы

## 1. Выполнены основные этапы анализа движения:

- Калибровка камер: Внутренняя калибровка с использованием шахматной доски и внешняя калибровка с использованием опорных точек в сцене.
  - Ошибка внутренней калибровки составила от 2.396 до 1.2 px, что превышает рекомендуемые значения (0.5 px).
  - Ошибка внешней калибровки составила от 9.996 до 14.319 px, что также выше рекомендованных значений (менее 1 см).

## 2. Оценка позы:

- Использование модели HALPE\_26 для обнаружения ключевых точек тела.
- Процесс оценки позы занял около 6 часов и 28 минут (процессор CPU).

## 3. Синхронизация видеоданных:

- Автоматическая синхронизация на основе скорости ключевых точек (*RWrist*).
  - Смещения между камерами: -30, 6, и 46 кадров с корреляцией от 0.25 до 0.38.

## 4. Ассоциация персон и триангуляция:

- Средняя ошибка перепроекции для точки Neck составила 11.1 px (~ 25.5 мм).
- Средняя ошибка перепроекции для всех точек на всех кадрах составила 6.9 px (~ 15.9 мм).
- В среднем пришлось исключить 1.93 камеры для достижения порога ошибок.

## 5. Фильтрация и интеграция с OpenSim:

- Применён фильтр Баттерворта низкой частоты 4-го порядка с частотой среза 6 Гц.
- Получены масштабированные модели и данные угловых движений суставов в формате *.mot*.

# Рекомендации

## 1. Улучшение качества калибровки:

- Увеличение количества изображений для внутренней калибровки:
  - Рекомендуется использовать не менее 10-15 различных позиций шахматной доски для более точного определения параметров камеры.
- Использование большего количества опорных точек для внешней калибровки:
  - Минимум 10 точек на каждой камере для повышения точности внешних параметров.
  - Убедитесь, что точки распределены по всей сцене, чтобы уменьшить искажения.

## 2. Оптимизация процесса оценки позы:

- Выбор более подходящих моделей:
  - Рассмотрите использование RTMPose или других моделей с более высокой точностью и скоростью.
- Настройка параметров обработки:
  - Отключите `display_detection` и `save_video` для ускорения процесса.
  - Увеличьте значение `det_frequency` для снижения нагрузки на вычисления.

## 3. Улучшение синхронизации видеоданных:

- Ручная синхронизация:
  - Если автоматическая синхронизация недостаточно точна, рассмотреть возможность использования звукового сигнала или вспышки света для ручной синхронизации.
- Улучшение освещения и условий съёмки:
  - Убедитесь, что освещение равномерное и отсутствуют блики, которые могут усложнить обнаружение ключевых точек.

# Заключение

Стажировка позволила получить практические навыки в области компьютерного зрения и машинного обучения, а также в разработке систем для анализа движений спортсменов.

Были достигнуты результаты в области детекции поз, синхронизации видео и визуализации движений.

Рекомендуется продолжить работу над улучшением качества калибровки камер и оптимизацией алгоритмов для повышения скорости обработки видео.







**Спасибо за внимание!**

