

Отчет об исследовании

Трекинг людей
по нескольким синхронизированным
камерам

Рылов Арсений Сергеевич
23.06.2019



Оглавление

Постановка задачи

Исходные данные

Описание готового решения

Ограничения решения

Проблемы

Проведенные эксперименты

Планы по улучшению решения

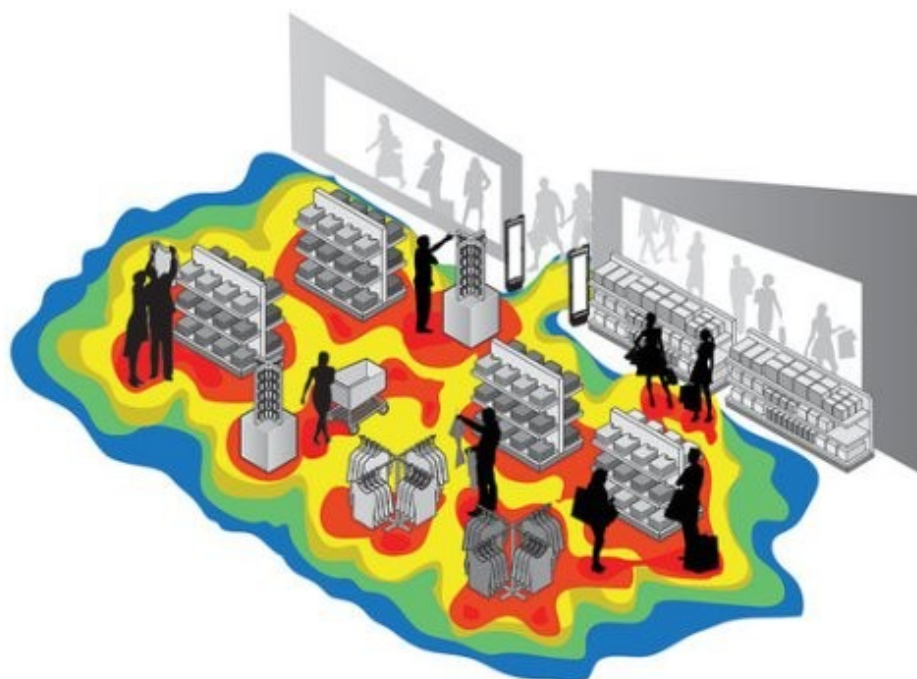
Выводы

Постановка задачи

Для осуществления контроля покупок и автоматизации магазинов самообслуживания без продавца необходимо в реальном времени **отслеживать перемещения каждого человека** по магазину самообслуживания от момента входа до завершения визита. Каждому человеку необходимо **присвоить** порядковый номер **ID**, к которому будет приписываться набор продуктов из его корзины.

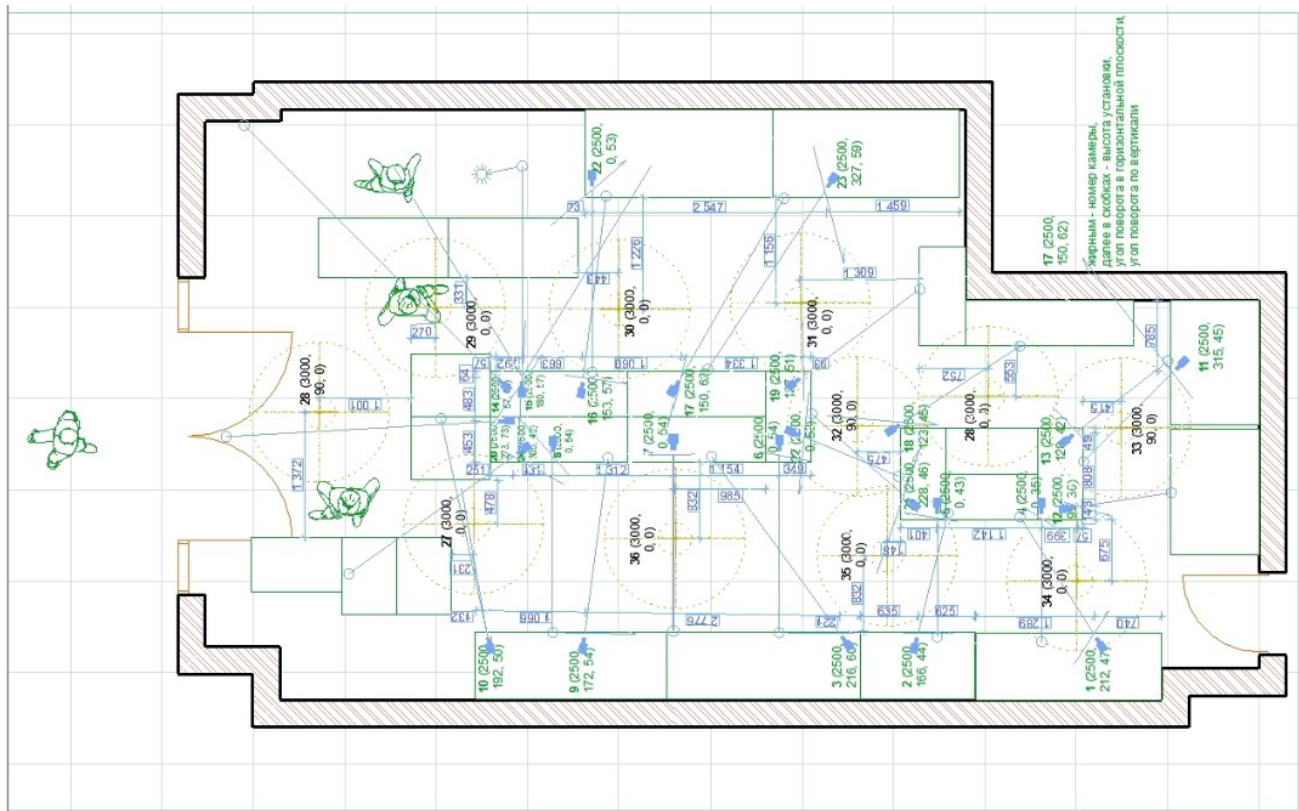
Трекинг – часть более общей задачи: **сверки** того, что взял человек по видео и того, что прошло по кассе. По итогам сверки можно получать сигналы и пресекать воровство.

Дальнейшие применения – отслеживание очередей, мест остановки, изучение предпочтений покупателей и индивидуальные предложения, heat map на плане помещения.

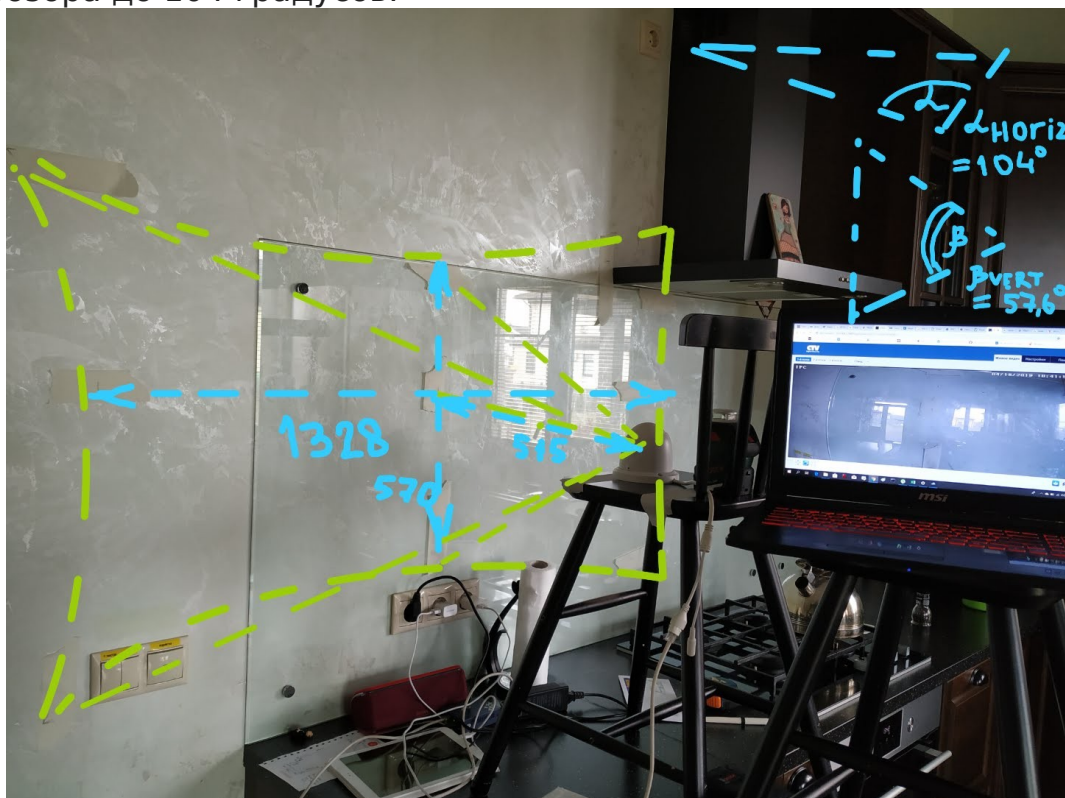


Исходные данные

Два магазина – 8 и 36 широкоугольных камер.



Углы обзора до 104 градусов.



Камеры, чаще всего, перекрывают друг друга (Это важно для решения!). Слепых зон нет. Синхронизация камер временно отсутствует: задержка видео до 12 секунд на некоторых камерах. Пока не устранены непостоянные лаги (до нескольких секунд - важно!)

В крупном магазине – более 100 визитов в день.

Одновременно в магазине находятся не более 15 человек.

По мере работы над задачей двигался “от простого к чуть более сложному”. От готового алгоритма трекинга встроенного в OpenCV для одной фигуры, к сравнению по фичам и реидентификации.

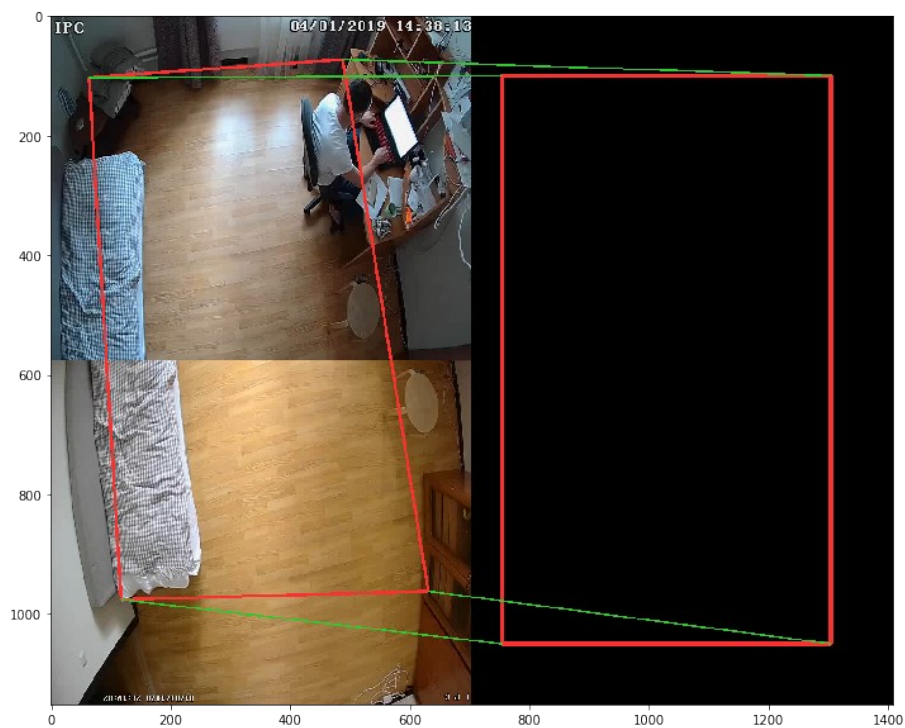
В итоге полученный алгоритм обладает некоторыми ограничениями, которые не позволяют эффективно работать со своими данными.

Пришлось брать видео для обкатки алгоритма с сайта соревнований по трекингу многих людей на одной камере. Имитации нескольких камер достиг, разрезая исходное видео на 3 картинки и неправильно склеивая их. <https://motchallenge.net/data/MOT17/>

Описание готового решения

Решение происходит в несколько этапов.

1. Разложение всех видео-поток на кадры.
2. Склейка из соответствующих кадров мозайки для визуального контроля.
3. Детекция людей. Один класс. Настройка confidence для улавливания слабых детекций, но при этом для отсутствия ложных срабатываний. Первый кадр с детектированным человеком инициализирует трекер – создает объект или объекты.
4. Non-maxima suppression. Передает далее массив баундинг боксов с неперекрывающимися детекциями. Параметр – threshold.
5. Поскольку зоны наблюдения камер перекрываются, можно сделать перспективные проекции изображений на план помещения, чтобы иметь общую систему координат.



6. Извлечение фичей для мэтчинга. Оставил после всех экспериментов единственную фичу – координаты баундинг боксов.

7. Сравнение детекций между соседними кадрами по принципу близости векторов фичей.

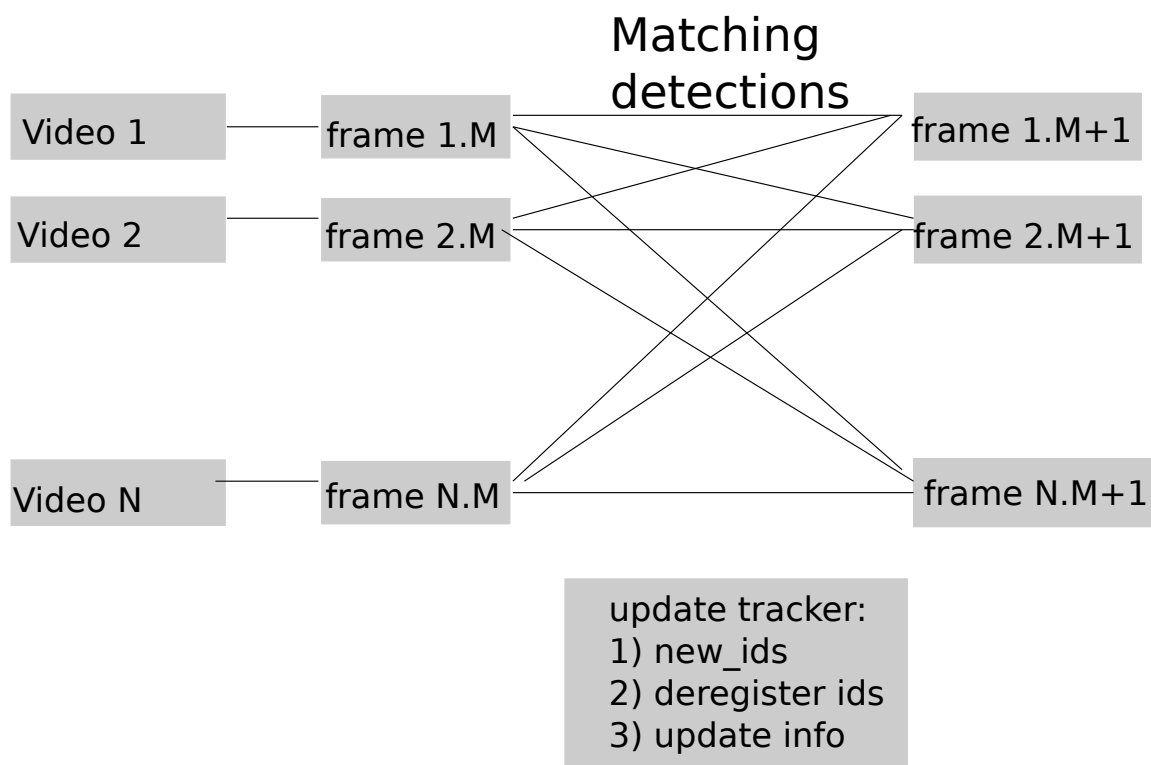


- В случае, когда очередная детекция далека от всех на предыдущем кадре – создается новый объект.

- В случае, когда детекции с прошлого кадра нет соответствия на текущем – объект трекинга пропадает с радара и помечается как исчезающий вид. После определенного количества кадров исчезнувший объект перестает участвовать в сравнении (безнадёжно потерян или ушел)
- Все остальные объекты трекинга обновляются новыми детекциями



Упрощенная схема работы алгоритма



Ограничения решения

- 1) Зоны охвата камер перекрываются. Нет мертвых зон.
- 2) Видео синхронизированы
- 3) Детектор работает уверенно. Нет ракурсов, при которых детекция теряется
- 4) нет перекрывающихся детекций

Метрики

Метрика качества трекера неотделима от корректности детектора.

1) Accuracy = TP/N , $17/35=48\%$

2) Rough accuracy = $1-abs(IDs_nmb-N)/N$,

IDs_nmb – число объектов созданных трекером.

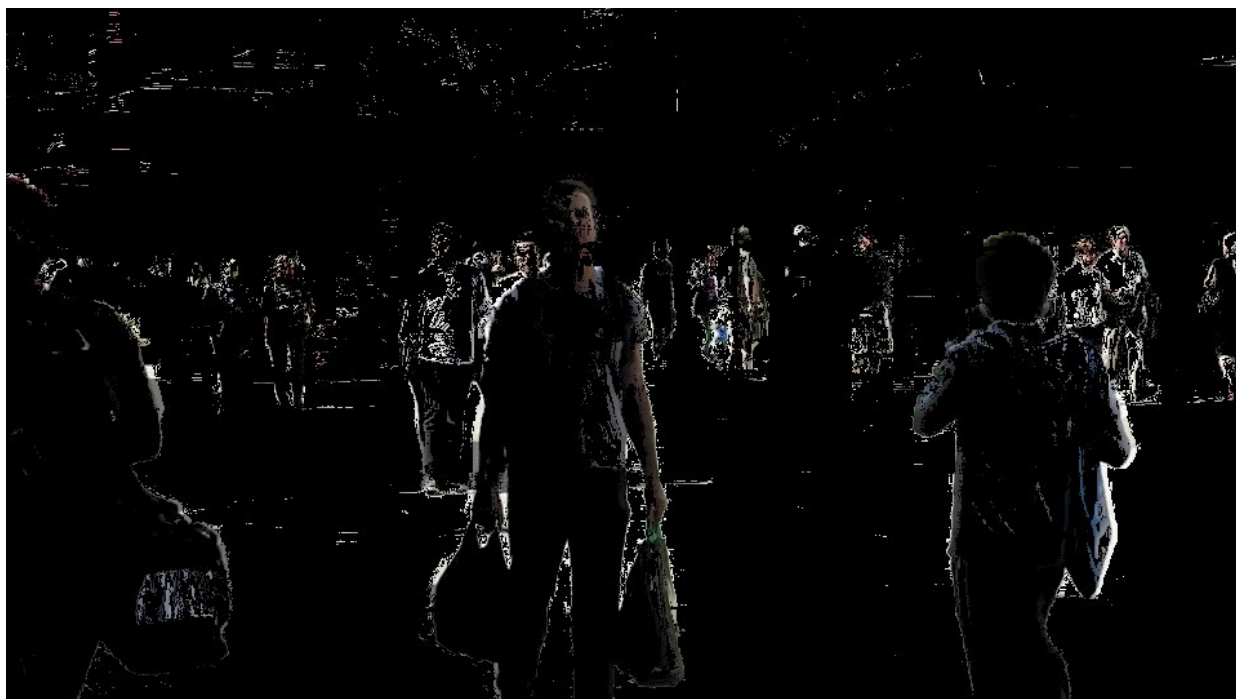
$1-(51-35)/35=51\%$

Проблемы

1. Разброс углов установки камер в вертикальной плоскости. От 0 до 60 градусов к вертикали
2. Люди очень по разному выглядят с разных ракурсов
3. Перекрывание детекций
4. Влияние фона на фичи
5. Отражения на стеклянных поверхностях
6. Ограниченность вычислительных ресурсов: две нейросети изначально не хотелось

Проведенные эксперименты

1. Восстановление **скелета человека Alpha Pose**. **Плюсы**: можно более точно спроектировать человека на плоскость пола. **Минусы**: шумы и ложные срабатывания. **Вывод**: сложно дообучать без Motion Capture студии



4. **Сокращение окна** баундинг бокса, для уменьшения влияния фона на фичи. **Вывод.** Немного улучшает. Можно дальше использовать.

Планы по улучшению решения

1. Применить **re-identification** или **фичи** из него. Преимущество, что потерянный из виду человек должен восстановиться.



2. 2. Применить **re-identification для видео** - склеивание локальных треков (на отдельной камере) в один глобальный

3. Учесть **близость камер**, скорость человека при мэтчинге.
Например, находясь в зоне камеры 2, человек в следующий момент может попасть в камеры 1 или 3. Таким образом, можно сравнивать не все со всеми детекции, а только некоторое подмножество.
4. **дообучить YOLO** детектировать людей на видах сверху
5. Попробовать восстановить **стереоскелет человека**, используя несколько камер и работать дальше в 3D.

Выводы

За время работы над дипломом мне удалось овладеть некоторыми полезными приемами по обработке изображений и видео.

Для решения проблемы перекрывающихся детекций нужно обучать детектор и реидентификацию на своих данных.

Список источников

- <https://www.pyimagesearch.com/2018/07/23/simple-object-tracking-with-opencv/>
- People Detection in Fish-eye Top-views. Meltem Demirkus, Ling Wang, Michael Eschey, Herbert Kaestle and Fabio Galasso
- Intelligent multi-camera video surveillance: A review. 2012 Xiaogang Wang
- Unsupervised Person Re-identification by Deep Learning Tracklet Association https://www.youtube.com/watch?v=-ljltl_arX8
- Person Re-identification: Past, Present and Future. Liang Zheng, Yi Yang, and Alexander G. Hauptmann