ЛЕКЦИЯ № 14 · Проект MATLAB IoT

Raspberry Pi, веб-камеры и ThingSpeak для анализа трафика на оживлённой трассе. Продолжение

Бельгибаев Бауржан Абдрахимович
Д.т.н., доцент
Кафедра «Искусственный интеллект и big data»
ОП 6B07113 «Интеллектуальные системы управления»

Цель лекции

Углубить навыки проектирования IoT-систем видеоаналитики транспортного потока: изучить устойчивое обнаружение и сопровождение транспортных средств, подсчёт пересечений контрольной линии, оценку интенсивности и скорости, обработку ошибок, передачу агрегированных показателей в ThingSpeak и оценку точности алгоритмов в MATLAB.

План лекции

1. Повторение архитектуры и постановка задачи
2. Подготовка видеопотока и геометрическая калибровка
3. Детектирование автомобилей в изменяющихся условиях
4. Сопровождение объектов и идентификаторы треков
5. Подсчёт пересечений контрольной линии
6. Интенсивность, плотность и средняя скорость потока
7. Вычислительные ограничения Raspberry Pi
8. Публикация агрегированных показателей в ThingSpeak
9. Аналитика MATLAB и графики загруженности
10. Оценка качества детекции и подсчёта
11. Сценарии отказов, безопасность и приватность
12. Сквозной воспроизводимый пример

1 Повторение архитектуры и постановка задачи

В лекции №13 рассматривался общий принцип системы «веб-камера — Raspberry Pi — алгоритм обработки кадров — ThingSpeak — MATLAB». Настоящая лекция посвящена переходу от обнаружения объектов на отдельных изображениях к устойчивому подсчёту транспортных средств и получению измеримых характеристик потока.

Видеокамера фиксирует последовательность кадров I(x,y,t). На устройстве выполняются предварительная обработка, выделение объектов, связывание обнаружений между соседними кадрами и регистрация событий пересечения линии. Затем формируются агрегированные показатели за фиксированные временные интервалы.

Для учебного проекта определим число проехавших автомобилей N за интервал Δt, интенсивность q=N/Δt и число ошибок подсчёта относительно эталонной ручной разметки. Отдельно исследуем влияние частоты кадров, освещения и пропусков изображения.

1

Повторение архитектуры и постановка задачи (продолжение)

Необходимо различать количество видимых автомобилей в кадре и количество уникальных автомобилей, прошедших контрольную линию. Первое является мгновенной оценкой занятости сцены, второе — событием движения. Суммирование числа объектов на всех кадрах приводит к многократному учёту одних и тех же машин.

2 Подготовка видеопотока и геометрическая калибровка

Качество видеоаналитики зависит от положения камеры, разрешения, экспозиции, устойчивости крепления и выбора области интереса ROI. В кадре следует выделить дорожное полотно и исключить участки, где движение не относится к исследуемому потоку.

Перспективное искажение приводит к тому, что одинаковые по размеру автомобили выглядят по-разному в зависимости от расстояния до камеры. Для подсчёта пересечений линии метрическая калибровка может не потребоваться, но для оценки скорости по пиксельным смещениям она обязательна.

При наличии опорных точек дорожной плоскости можно оценить проективное преобразование (гомографию) и перейти от координат изображения к координатам на плоскости дороги. Матрица H задаёт соотношение $[u,v,1]^T \sim H[x,y,1]^T$. Для реальной камеры H определяют по известным геометрическим соответствиям.

Частота кадров f определяет интервал $\Delta t=1/f$. При низкой частоте быстрый автомобиль может переместиться на большое расстояние между соседними кадрами и потерять соответствие с треком. Выбор частоты — компромисс между точностью и вычислительной нагрузкой.

2 Подготовка видеопотока и геометрическая калибровка (продолжение)

Синхронизация времени необходима, если показатели сопоставляются с данными нескольких камер или внешних датчиков. В журнале желательно сохранять временную метку кадра, номер кадра и признак корректности обработки.

3 Детектирование автомобилей в изменяющихся условиях

Один из базовых подходов — вычитание фона: $D_t(x,y)=|I_t(x,y)-B_t(x,y)|$, где B_t — оценка фонового изображения. Пиксели с $D_t>\tau$ относят к движущимся областям. Метод сравнительно прост, но чувствителен к теням, дождю, изменению освещённости и движению камеры.

Для выделения связных областей применяют пороговую обработку, морфологические операции и фильтрацию по площади. Размер минимального объекта должен соответствовать перспективе и разрешению; слишком высокий порог удалит дальние автомобили, а слишком низкий создаст ложные объекты.

Современный альтернативный подход — детекторы на основе нейронных сетей. Они могут лучше разделять классы «автомобиль», «грузовик», «автобус», однако требуют подходящей модели, вычислительных ресурсов и проверки качества на конкретной сцене.

При частичном перекрытии двух автомобилей один детектор может выдать единую область, а при изменении освещения — несколько ложных областей. Поэтому детектирование следует оценивать отдельно от сопровождения и подсчёта.

В учебном MATLAB-проекте допускается сначала использовать заранее подготовленную таблицу обнаружений, чтобы отладить логику пересечений независимо от модели компьютерного зрения.

4 Сопровождение объектов и идентификаторы треков

Сопровождение (tracking) связывает обнаружения одного автомобиля в последовательных кадрах и присваивает ему устойчивый идентификатор track_id. Без идентификаторов невозможно надёжно отличить новый автомобиль от уже подсчитанного.

Простейшее связывание основано на расстоянии между центрами ограничивающих прямоугольников. Для обнаружений i и j расстояние $d_{ij}=\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}$. Соответствие допускается, если d_{ij} меньше порога, согласованного со скоростью движения и интервалом кадров.

Более устойчивые методы используют предсказание положения, например фильтр Калмана, и алгоритм назначения соответствий. При пропуске детекции трек может временно сохраняться, но слишком долгий пропуск увеличивает риск ошибочного связывания.

Трек должен содержать ID, историю координат, время последнего обнаружения, направление движения и флаг «уже подсчитан». Последний признак особенно важен для исключения повторного учёта при колебаниях координаты около линии.

При анализе нескольких полос нужно учитывать направление и принадлежность к полосе. Автомобиль, изменивший полосу, не должен автоматически считаться новым объектом.

5 Подсчёт пересечений контрольной линии

Контрольная линия задаётся двумя точками $A(x_1,y_1)$ и $B(x_2,y_2)$. Для точки $P(x,y)$ знак ориентированной функции $s(P)=(x_2-x_1)(y-y_1)-(y_2-y_1)(x-x_1)$ показывает, с какой стороны прямой находится центр объекта.

Событие пересечения можно зарегистрировать, если для одного track_id знак $s(P)$ изменился между двумя последовательными корректными положениями. Для определения направления проверяют порядок знаков. В реальном алгоритме также следует проверять, что траектория пересекает именно отрезок линии, а не его бесконечное продолжение.

Шум координат около линии вызывает повторные изменения знака. Для устойчивости вводят мёртвую зону, подтверждение движения в заданном направлении и флаг counted. После регистрации одного пересечения трек не должен считаться повторно.

Пример: при горизонтальной линии y=240 автомобиль движется сверху вниз. Событие возникает при y_prev<240 и y_curr>=240. Если tracked ID уже находится в множестве подсчитанных, новое событие игнорируется.

5 Подсчёт пересечений контрольной линии (продолжение)

При пересечении линий в двух направлениях удобно вести два счётчика: N_in и N_out. Их разность может использоваться для приблизительной оценки накопления автомобилей в контролируемой зоне при условии полного покрытия всех входов и выходов.

```
% MATLAB: учебная логика пересечения для одного трека
lineY = 240;
y = [180 205 230 245 265 280];
counted = false;
count = 0;
for k = 2:numel(y)
    crossed = y(k-1)<lineY && y(k)>=lineY;
    if crossed && ~counted
        count = count+1;
        counted = true;
    end
end
fprintf('Пересечений: %d\n',count);
```

6 Интенсивность, плотность и средняя скорость

Интенсивность движения q определяется числом транспортных средств N, прошедших контрольное сечение за время Δt: q=N/Δt. Если N=48 автомобилей за 10 минут, то q=4,8 авт./мин или 288 авт./ч. Единицы измерения необходимо указывать в графиках и отчётах.

Плотность k характеризует число автомобилей на единицу длины дороги, например авт./км. Для приближённой модели однородного потока справедливо фундаментальное соотношение q=k·v̄, где v̄ — средняя пространственная скорость. Это соотношение не следует применять без проверки предположений к неоднородным и многополосным потокам.

Для оценки скорости по видео требуется метрическая калибровка. Если автомобиль прошёл известное расстояние L между двумя виртуальными линиями за время Δt, то v=L/Δt. Для перевода м/с в км/ч результат умножают на 3,6.

Например, L=25 м, Δt=2 с, тогда v=12,5 м/с=45 км/ч. Ошибка в определении расстояния и времени напрямую влияет на оценку скорости. При отсутствии калибровки корректно сообщать лишь скорость в пикселях в секунду, не называя её физической скоростью автомобиля.

Агрегирование по интервалам 1, 5 или 15 минут сглаживает случайные колебания, но может скрывать кратковременные заторы. Размер окна выбирают по цели исследования.

7 Вычислительные ограничения Raspberry Pi

Обработка видео требует ресурсов процессора, оперативной памяти и системы хранения. Нагрузка зависит от разрешения кадра, частоты обработки, сложности детектора и количества одновременно отслеживаемых объектов.

Для оценки производительности измеряют время обработки одного кадра t_proc и фактическую частоту f_eff≈1/t_proc при последовательной обработке. Если камера выдаёт 30 кадров/с, а алгоритм обрабатывает только 8 кадров/с, необходимо определить политику пропуска кадров или оптимизировать вычисления.

Типовые способы оптимизации: уменьшение ROI, масштабирование кадра, пакетная обработка, применение более лёгкой модели и перенос тяжёлой аналитики на отдельный сервер. Каждый способ может снижать точность, поэтому изменения следует проверять экспериментально.

Для длительного наблюдения важны энергопотребление, температурный режим, устойчивость питания и объём локального журнала. В учебном отчёте полезно приводить фактическую частоту обработки, загрузку процессора и долю потерянных кадров.

7

Вычислительные ограничения Raspberry Pi (продолжение)

Сетевой обмен не следует выполнять синхронно на каждом кадре: задержка запроса может остановить обработку. Более надёжная архитектура разделяет обработку видео и публикацию агрегированных результатов в отдельные задачи.

8 Публикация показателей в ThingSpeak

ThingSpeak используется не как видеохранилище, а как платформа временных рядов для агрегированных результатов. Например, Field 1 — количество автомобилей за интервал, Field 2 — интенсивность в авт./ч, Field 3 — оценка средней скорости в км/ч, Field 4 — показатель качества обработки.

После завершения временного окна Raspberry Pi формирует запись и отправляет её в канал через API. Частота публикации должна соответствовать действующим ограничениям аккаунта и сервиса. Ключ записи необходимо хранить безопасно и не включать в открытый код.

Важно сохранять время начала и конца интервала. Если при сбое сети агрегаты отправляются позднее, время загрузки в облако не равно времени наблюдения. Следует проверить возможности API по сохранению исходных временных меток.

Если за интервал не поступило кадров, нулевое количество автомобилей нельзя автоматически считать достоверным: отсутствие наблюдений отличается от наблюдаемого нулевого потока. Полезно передавать отдельный признак доступности измерения.

При построении графика интенсивности в MATLAB необходимо учитывать пропущенные интервалы и не соединять их без пояснения как обычные наблюдения.

8

Публикация показателей в ThingSpeak (продолжение)

```
% MATLAB: пример формирования агрегированных полей
count10min = 48;
windowMinutes = 10;
flowPerHour = count10min*60/windowMinutes;
meanSpeed = 45;  % учебное значение, км/ч
payload = struct('field1',count10min, ...
                 'field2',flowPerHour, ...
                 'field3',meanSpeed);
disp(jsonencode(payload));
% Отправка требует своего канала и API-ключа.
```

9 MATLAB-аналитика загруженности дороги

После получения агрегированных данных можно строить графики интенсивности по времени, сравнивать интервалы, рассчитывать средние и пиковые значения, а также выделять периоды роста нагрузки. Для корректного сравнения интервалы должны иметь одинаковую продолжительность либо значения следует нормировать по времени.

Скользящее среднее q̄_t=(1/m)Σq_(t−i) позволяет увидеть общий тренд. Однако сглаживание не заменяет исходные данные и может скрыть кратковременные пики. В отчёте желательно показывать оба ряда.

Для сравнения утренних и вечерних интервалов можно вычислить медиану, квартильный размах и долю окон с превышением заданной интенсивности. Выбор порога должен быть связан с характеристиками конкретной дороги, а не взят произвольно.

При анализе нескольких дней следует учитывать выходные, погоду, дорожные работы и особенности расположения камеры. Сходство графиков не доказывает, что условия наблюдения одинаковы.

9 MATLAB-аналитика загруженности дороги (продолжение)

MATLAB Online может использоваться для анализа экспортированного CSV-файла или доступных записей ThingSpeak. Прямой доступ к видеоустройству из облачной среды не является обязательным для аналитической части.

```
% MATLAB: визуализация учебных 10-минутных окон
window = (1:12).';
N = [18;22;30;39;48;55;52;46;41;35;28;21];
q = N*6;   % авт./ч при 10-минутных окнах
smoothQ = movmean(q,3);
figure; plot(window,q,'-o','LineWidth',1.3); hold on;
plot(window,smoothQ,'-','LineWidth',1.8);
grid on; xlabel('Номер 10-минутного интервала');
ylabel('Интенсивность, авт./ч');
legend('Исходные данные','Скользящее среднее');
title('Учебная динамика транспортного потока');
```

10 Оценка качества детекции и подсчёта

Для проверки алгоритма требуется эталонная разметка видеозаписи: перечень автомобилей и моменты их пересечения контрольной линии. Эталон создают вручную либо независимым проверенным способом. Без него нельзя объективно утверждать о точности подсчёта.

Для детекции объектов используются показатели Precision=TP/(TP+FP) и Recall=TP/(TP+FN), где TP — верные обнаружения, FP — ложные обнаружения, FN — пропущенные объекты. F1=2PR/(P+R) объединяет точность и полноту.

Для подсчёта потока полезна абсолютная ошибка E_abs=|N_pred−N_true| и относительная ошибка E_rel=|N_pred−N_true|/N_true·100 %. При N_true=0 относительная ошибка в таком виде не определена, поэтому требуется отдельное правило отчётности.

Ошибки сопровождения проявляются в смене ID, потере трека и двойном подсчёте. Даже высокая точность детекции отдельных кадров не гарантирует правильного числа уникальных автомобилей.

Эксперимент желательно проводить в разных условиях: дневное и вечернее освещение, низкая и высокая интенсивность, частичное перекрытие объектов, дождь и пропуски кадров. Результаты представляют отдельно по сценариям, а не только одной общей цифрой.

10 Оценка качества детекции и подсчёта (продолжение)

```
% MATLAB: сравнение с ручным подсчётом
Ntrue = [20 35 48 52 41 25];
Npred = [19 34 46 54 40 26];
absErr = abs(Npred-Ntrue);
relErr = 100*absErr./Ntrue;
MAE = mean(absErr);
MAPE = mean(relErr);
fprintf('MAE: %.2f автомобиля\n',MAE);
fprintf('Средняя относительная ошибка: %.2f %%\n',MAPE);
```

11

Отказы, информационная безопасность и приватность

Основные источники отказов — отключение камеры, недостаток питания, переполнение накопителя, снижение частоты обработки, плохая видимость и потеря сети. Каждое событие должно регистрироваться, чтобы аналитик мог отличить реальное снижение потока от технического сбоя.

При невозможности обработать кадр система не должна молча увеличивать или обнулять счётчик. Рекомендуется фиксировать долю обработанных кадров и состояние качества наблюдения. Для пропущенных интервалов вводят отдельную отметку «данные отсутствуют».

Видеозаписи могут содержать лица и государственные регистрационные знаки. Для учебного мониторинга предпочтительно передавать в облако только обезличенные агрегированные показатели, а хранение исходного видео ограничивать необходимым сроком и правами доступа.

Следует соблюдать применимые требования законодательства о персональных данных, правила установки камер и разрешения владельца площадки. Публикация видеоматериалов без соответствующих оснований может создавать правовые риски.

Ключи ThingSpeak хранят отдельно от публичного исходного кода. Используют HTTPS, разграничение доступа и журналирование ошибок. Удалённые команды не должны подменять локальные механизмы безопасной работы оборудования.

12

Сквозной воспроизводимый пример и план эксперимента

Рассмотрим учебный набор данных для одной полосы дороги. В течение 12 последовательных 10-минутных интервалов получены количества автомобилей N=[18,22,30,39,48,55,52,46,41,35,28,21]. Это искусственные значения, предназначенные для проверки MATLAB-алгоритма, а не реальные результаты наблюдения.

Для каждого интервала интенсивность рассчитывается как q=N·60/10=6N авт./ч. Например, для N=55 получаем q=330 авт./ч. Далее строятся график q и скользящее среднее по трём интервалам.

На втором этапе вводятся эталонные и автоматические результаты подсчёта, вычисляются MAE и относительная ошибка. На третьем этапе исследуется влияние пропусков кадров и неверного сопоставления треков.

При наличии Raspberry Pi и веб-камеры учебный массив заменяется фактическими событиями пересечения линии. Сбор выполняется локально, а ThingSpeak принимает только итоговые значения за окна наблюдения.

12

Сквозной воспроизводимый пример и план эксперимента (продолжение)

Отчёт должен включать схему системы, параметры камеры, описание ROI и линии, алгоритм трекинга, временные интервалы, графики интенсивности, оценку точности, перечень отказов и ограничения эксперимента.

Критерий успешности — не просто наличие графика в облаке, а доказанная корректность подсчёта по независимому эталону, воспроизводимость вычислений и явное указание условий наблюдения.

3 Заключение

В лекции рассмотрен переход от видеодетекции к количественному анализу транспортного потока: сопровождение объектов, исключение повторного учёта, пересечение контрольной линии, вычисление интенсивности и скорости, агрегирование данных и публикация в ThingSpeak. Показаны подходы к оценке ошибок, оптимизации вычислений на Raspberry Pi и обеспечению приватности. MATLAB используется для воспроизводимого расчёта показателей и визуального анализа временных рядов.

Контрольные вопросы (1/2)

1. Чем число автомобилей в кадре отличается от числа пересечений линии?
2. Какова роль области интереса ROI?
3. Почему перспектива влияет на оценку скорости?
4. Что такое гомография?
5. Как частота кадров влияет на сопровождение?
6. Как работает метод вычитания фона?
7. Почему тени создают ложные обнаружения?
8. Что такое track_id?
9. Для чего нужен фильтр Калмана в трекинге?
10. Как определяется расстояние между центрами объектов?
11. Почему требуется флаг counted?
12. Как обнаружить пересечение контрольной линии?
13. Как учитывать направление движения?
14. Что такое интенсивность транспортного потока?
15. Как перевести авт./мин в авт./ч?
16. Что означает соотношение $q=k \cdot v$?

Контрольные вопросы (2/2)

17. Какие условия нужны для оценки скорости в км/ч?

18. Как оценить производительность Raspberry Pi?

19. Почему не следует отправлять запрос на каждый кадр?

20. Какие показатели целесообразно передавать в ThingSpeak?

21. Чем время измерения отличается от времени загрузки?

22. Как используется movmean?

23. Что означают Precision и Recall?

24. Как вычисляется относительная ошибка подсчёта?

25. Какие меры приватности нужны при видеомониторинге?

Д Задания для самостоятельной работы

1. Реализуйте в MATLAB подсчёт пересечений горизонтальной линии для пяти траекторий с уникальными ID.

2. Для 12 десятиминутных интервалов вычислите интенсивность в авт./ч и постройте график со скользящим средним.

3. Разработайте алгоритм исключения повторного подсчёта при колебании объекта около линии.

4. Рассчитайте скорость автомобиля для дистанций 20, 25 и 30 м и времени прохождения 1,5; 2,0 и 2,5 с.

5. Сравните ручной и автоматический подсчёт по шести интервалам; вычислите MAE и относительные ошибки.

6. Предложите архитектуру Raspberry Pi — ThingSpeak — MATLAB с буферизацией и защитой персональных данных.

Рекомендуемая литература

• Силлабус дисциплины «Проект MATLAB IoT», тема лекции №14.

• MathWorks. MATLAB Documentation — https://www.mathworks.com/help/matlab/

• MathWorks. Computer Vision Toolbox Documentation — https://www.mathworks.com/help/vision/

• MathWorks. ThingSpeak Documentation — https://www.mathworks.com/help/thingspeak/

• Raspberry Pi Documentation — https://www.raspberrypi.com/documentation/