

Система «Умная камера» на Raspberry Pi 3

Аннотация

В данной работе описан процесс создания прототипа системы «Умная камера» на Raspberry Pi 3. Для анализа требований потенциальных пользователей системы было использовано глубинное интервью с владельцами среднего и малого бизнеса. Результат опроса показал, что наиболее интересной функцией для потенциальных пользователей является определение интенсивности посетителей, вследствие чего была поставлена задача определения количества людей на изображении. Для решения данной задачи использовалась модель YOLO Tiny версии 3, обученная на наборе данных SCUT-HEAD и реализованная на фреймворке машинного обучения Tensorflow и Tensorflow Lite. Скорость распознавания с изображения для модели Tensorflow составила 2 секунды, для Tensorflow Lite – 1 секунду.

Ключевые слова: smart cameras, object detection, image segmentation, image processing, software prototyping, prototypes, data visualization, monitoring, Raspberry Pi 3, IOT.

Введение

На сегодняшний день тренд в мире электроники заключается в том, что стоимость одноплатных компьютеров падает, а их производительность растет. Это позволяет уже сейчас использовать данные устройства для реализации сложных алгоритмов. Но при этом на сегодняшний день на рынке не существует доступных устройств, способных предоставлять владельцам малого и среднего бизнеса оффлайн-услуги по дополнительной аналитике посетителей их заведений, основанной на решении задачи распознавания объектов с помощью автоматизированных средств.

Считается, что лучшее качество в задачах распознавания объектов показывают алгоритмы, основанные на нейронных сетях. При этом нужно учитывать, что такие нейронные сети требуют, как правило, больших вычислительных мощностей. Для того чтобы данную аналитику можно было реализовать на одноплатном компьютере, данный алгоритм нужно оптимизировать: ускорить вычисление нейронных сетей, используемых при решении задачи распознавания объектов на изображении.

Данного ускорения можно добиться путем применения одного или нескольких методов по оптимизации нейронных сетей: представление сети в целочисленной форме или уменьшение количества знаков после запятой в параметрах сети, использование сети с меньшим количеством параметров для обучения на ответах другой нейронной сети, которая имеет больше параметров, метода последовательных тернарных решений и других.

Целью данной работы является создание системы проведения продвинутой аналитики аудитории различных предприятий массового обслуживания и автоматизации существующих сейчас процессов. Для достижения указанной выше цели нами были поставлены следующие задачи: опросить владельцев малого и среднего бизнеса, чтобы выяснить, какие параметры системы будут наиболее актуальными, проанализировать существующие аналоги разрабатываемой системы и подходы к решению задачи определения количества людей в помещении, реализовать прототип системы.

Обзор существующих систем

На данный момент существуют различные системы, осуществляющие подсчет людей в общественных местах и аналитику этих посещений. Они отличаются по физическим и программным принципам подсчета, а также имеют различные возможности и функциональные особенности. Данные системы можно классифицировать следующим образом:

- системы, работающие по инфракрасному лучу;
- системы, работающие на тепловых датчиках;
- системы с использованием видеодатчиков.

В данной работе нами будут рассматриваться только последние, так как очевидно, что использование камеры в качестве датчика более перспективно и позволяет создать более функционально продвинутое устройство.

Одним из существующих на рынке решений является вертикальный видеосчетчик посетителей **Rstat Real-2D Light**. [1] Подсчет посетителей основан на видеосъемке. Для просмотра данных разработан веб-интерфейс в облаке.

Технические характеристики Rstat Real-2D Light:

Камера: нет данных;

Процессор: 8-ми ядерный, частота 3.4Ггц;

Питание: PoE (IEEE802/3af);

Память: встроенная карта памяти на 8Гб. Хранение информации на носителе до 5 лет;

Размер, форм-фактор: Вертикальный видеосенсор. 149x114x57 мм.

По описанию производителя, устройство поддерживает настройку маршрутов и линий подсчета, разных областей (в том числе области проходящих мимо людей, которых учитывать не нужно). Отмечена возможность подсчета сотен людей в зоне (однако при зоне видимости 6.5x4.8м., одновременное нахождение такого количества людей на указанной области представляется маловероятным).

Другим найденным на рынке решением является **Track Expert**. [2] Решение использует технологии компьютерного зрения для подсчета посетителей, определение пола и возраста, определения эмоции посетителя. Стоимость решения колеблется от 9900 (входит лишь подсчет людей) до 13900 (версия со всеми вышеперечисленными возможностями). В стоимость входит одно устройство.

Технические характеристики Track Expert: нет данных.

Устройство **Zen Count** [3] осуществляет подсчет посетителей на входе и выходе с помощью алгоритмов видеоаналитики. Видеосчетчик имеет встроенную камеру. В основе видеосчетчика лежит контроллер, передача данных и питание осуществляется через слаботочную сеть заказчика по обычной витой паре. Питание счетчика осуществляется через стандартный адаптер PoE. Данный видеосчетчик устанавливается над входом и направлен вертикально вниз. Входящий или выходящий клиент определяется алгоритмом видеоаналитики и формирует в базу данных событие о факте прохода. Датчиком охватывается площадь пола примерно 7 на 7 метров.

Технические характеристики Zen Count:

Камера: 5 Мп, 2592x1944; FullHD 30 fps, HD 60 fps;

Процессор: Broadcom BCM2835, 700МГц;

Питание: PoE (IEEE802/3af);

Память: нет данных;

Размер, форм-фактор: вертикальный видеосенсор.

На основании представленных технических данных можно сделать вывод, что система основана на одноплатном компьютере Raspberry Pi первого поколения. В качестве видеокамеры выступает Raspi CAM.

Таким образом, данное устройство реализует лишь функции подсчета посетителей без дополнительных аналитик. Стоимость одного устройства составляет 29490 рублей (так как охватываемая территория не очень большая, может потребоваться больше датчиков). Установка и монтаж – 6000, а сервисная поддержка – 1000 рублей ежемесячно.

И последним устройством, существующим в продаже, рассмотрим счетчик посетителей **MegaCount**. [4]

Технические характеристики MegaCount:

Камера: CMOS с поддержкой HDR;

Процессор: одноядерный DSP процессор;

Питание: PoE 48V;

Память: встроенная 8 Гб;

Размер, форм-фактор: вертикальный видеосенсор.

2D счетчик посетителей использует функцию распознавания перемещения относительно статичного фона. Устройство получает поток картинок с камеры, анализируя их и сравнивая, находит изменения относительно фона и вычисляет траектории движения. По своей сути устройство лишь считает проходы относительно некой точки. Предусмотрена возможность лишь количественной аналитики. Стоимость колеблется в районе 25000 рублей.

Было обнаружено еще несколько похожих систем, но для всех них можно справедливо отметить относительно высокую стоимость и то, что у большинства из них функции ограничиваются подсчетом входящих и выходящих посетителей и, соответственно, аналитика строится лишь на этих данных.

Нами предлагается разработка устройства, позволяющего вести более продвинутую аналитику посетителей, включая следующие возможности:

- подсчет людей в помещении (плюс исключение из аналитики персонала);
- гибкая настройка зон и маршрутов в помещении для подсчета;
- определение гендерной принадлежности посетителя;
- определение возрастной группы посетителя;
- определение эмоции человека в помещении.

В качестве основы предполагается использование одноплатного компьютера Raspberry Pi версии 3B+ или старше.

Обзор литературы

На данный момент существует два основных подхода для решения задачи определения количества людей в помещении: детекция объектов (object detection) и подсчет толпы (crowd counting).

Одной из самых первых статей по теме подсчета количества людей автоматизированными средствами является «A counting method of the number of passing people using a stereo camera». [5] В данной статье используются две камеры, смотрящие на разделяющую линию и считающие количество пройденных людей ровно по линии под камерами. Данный подход не дает полного понимания и надежности в оценки количества людей в помещении, т. к. в помещении должен быть только один вход и выход, а в общественных заведениях часто возникает обратная ситуация. Также проблемой является тот факт, что для удачной работы алгоритма люди должны проходить ровно по линии, а это значит, что слишком высокие или слишком низкие люди не будут видны данной камере. В статье «Counting people in crowds with a real-time network of simple image sensors» люди продолжают идею использования нескольких камер для проекции изображений в одну

плоскость и нахождения точного количества людей. Данный подход достаточно сложный, т. к. требует индивидуальной настройки под каждое помещение ввиду расчетов на основе геометрии данного помещения и большого количества камер, что тоже является минусом ввиду увеличения стоимости системы. [6] Также для обнаружения и отслеживания людей на пересеченной местности используются методы [7, 8, 9, 10], которые основаны на регрессии. Их цель состоит в том, чтобы изучить соответствие между низкоуровневыми характеристиками и количеством людей в толпе. Эти методы также являются специфичными, поскольку модель, построенная для одной местности, может быть применена только к ней. В последнее время с увеличением вычислительных мощностей, которые используются для построения моделей, и объемов данных для обучения все большую популярность для решения задач подсчета людей в толпе приобретают модели, основанные на нейронных сетях. В работе «Cross-scene crowd counting via deep convolutional neural networks» проблема подсчета толпы решается с помощью сверточной нейронной сети. Данный метод основан на оценки плотности и количества людей в толпе, что позволяет добиться более точных результатов. [11]

Обнаружение объектов – это компьютерная технология, связанная с компьютерным зрением и обработкой изображений, которая основывается на обнаружении экземпляров семантических объектов определенного класса (таких как люди, здания или автомобили) в цифровых изображениях и видео. Хорошо изученные области обнаружения объектов включают в себя обнаружение лиц и пешеходов. Обнаружение объектов находит применение во многих областях компьютерного зрения, включая поиск изображений и видеонаблюдение. Одной из первых работ по детекции объектов является статья «A general framework for object detection». В ней описывается подход, основанный на вейвлет-преобразованиях и методе опорных векторов для нахождения объектов на фотографии. [12] Однако качество данного подхода оставляло желать лучшего, в связи с чем данный алгоритм годился только для детекции лиц на изображении. В статье «Robust real-time object detection» используется метод Виолы-Джонса, который заключается в использовании изображений в интегральном представлении, использовании признаков Хаара и принципа сканирующего окна. [13] Вышеперечисленные методы хорошо работали только на лицах людей, но, для детекции людей в толпе, необходимо либо распознавать людей целиком, либо только головы. Именно поэтому необходимо обратить внимание в сторону более сложных моделей, основанных на нейронных сетях. В 2016 году модель YOLA [14] показала наилучший результат по соотношению скорость выполнения - качество показания. Модель основана на сверточной нейронной сети, которая по плотности принадлежности участков изображения к одному классу оценивает вероятность того, что

область принадлежит конкретному классу. Подход [15] основан на применении архитектуры нейронной сети Faster R-CNN, и на текущий момент он показывает наилучший результат по детектированию объектов, однако авторы используют видеокарты Nvidia Titan X GPU, а использование таких больших ресурсов чрезмерно для решения нашей задачи.

Анализ существующих наборов данных

Набор данных HollywoodHead dataset [16] содержит 369 846 человеческих голов, аннотированных в 224 740 видеокдрах из 21 голливудского фильма. Для создания аннотации была осуществлена ручная разметка видеодорожки: для каждой дорожки на нескольких ключевых кадрах вручную были выделены прямоугольники, ограничивающие головы. Ограничительные рамки на оставшихся кадрах были линейно интерполированы и затем проверены вручную на правильность. HollywoodHead dataset разделен на обучающую, валидационную и тестовую выборки, каждая из которых содержит различные фильмы. Таким образом, обучающая выборка содержит 216 719 кадров из 15 фильмов, валидационная выборка содержит 6 719 кадров из 3 фильмов, а тестовая выборка – 1302 кадра из оставшихся 3 фильмов. Данная выборка не содержит трудно различимых человеческих голов (например, из-за плохого освещения).

Набор данных Brainwash [17] – это набор данных, созданный для решения задачи обнаружения человеческих голов. Он также разделен на три выборки: обучающая выборка содержит 10 000 изображений, валидационная и тестовая около 500 изображений каждая. Все изображения получены с неподвижной камеры наблюдения в кафе-магазине. На сегодняшний день данный набор данных изъят из общего доступа.

Еще одним известным набором данных с помеченными человеческими головами является набор SCUT-HEAD [18], который включает в себя 4405 изображений, содержащих 111 251 аннотированных человеческих голов. Данный набор состоит из двух частей: PartA и PartB. PartA включает в себя 2000 изображений, которые были отобраны из видеонаблюдения в аудиториях университета. Данная часть содержит 67 321 помеченную голову. PartB содержит 2405 изображений, отсканированных из Интернета и содержащих 43 930 голов. Каждая видимая голова была помечена координатами x_{min} , x_{max} , y_{min} , y_{max} . В процессе аннотирования старались охватить всю голову, включая те её части, которые закрывает другой объект, но при этом стремились избежать захвата дополнительного фона. Обе части разделены на обучающую и тестовую выборки. Этот набор данных соответствует стандарту VOC.

Основная часть

При разработке данной системы мы использовали следующие методы:

1. Глубинное интервью с владельцами малого и среднего бизнеса;
2. Опрос владельцев малого и среднего бизнеса по методу Саати (парное сравнение альтернатив по каждому из критериев и парное сравнение критериев с точки зрения важности для поставленной цели);
3. Аналитический обзор возможных аналогов разрабатываемой системы и подходов к решению задачи определения количества людей в помещении.

Для глубинного интервью использовались следующие вопросы:

1. Какого рода заведение вы представляете? Опишите его.
2. Нужна ли вам дополнительная автоматизированная аналитика посетителей?
3. Какие аналитические данные вам были бы полезны: интенсивность посещения (количество людей в зависимости от времени), аналитика по полу посетителей, аналитика по возрасту, определение постоянных посетителей.
4. Если в предыдущем вопросе нет интересующего пункта, опишите его.
5. Какая модель оплаты для вас наиболее приемлема: разовая покупка коробочного решения или аренда системы?
6. Какой способ взаимодействия с системой для вас наиболее удобен: веб-интерфейс, мобильное приложение, чат-бот в мессенджере, email рассылка?
7. Желаемое крепление камеры в помещении?
8. Какую сумму вы готовы потратить при коробочном решении и при ежемесячной аренде?

Для оценки того, какой способ взаимодействия с системой наиболее удобен для потенциальных клиентов, можно использовать матрицу попарных сравнений. Матрица попарных сравнений является инструментом количественного анализа при принятии решений, который позволяет команде значительно упростить нередко такой запутанный процесс определения приоритетов. Также данный метод подойдет для определения приоритетного критерия из следующего набора критериев системы: цена системы, надежность устройства, безопасность, удобство взаимодействия, наличие сервисной поддержки, эргономичность корпуса, дополнительные функции аналитики.

Для данного метода необходимо составить матрицу, где a_{ij} – отношение критерия i к критерию j . $a_{ji} = \frac{1}{a_{ij}}$, $a_{ii} = 1$. Для нормировки матрицы находится сумма элементов каждого столбца: $S_j = a_{1j} + a_{2j} + \dots + a_{nj}$. После этого все элементы матрицы делятся на сумму элементов соответствующего столбца: $A_{ij} = \frac{a_{ij}}{S_j}$. Затем вычисляется среднее

значение для каждой строки – полученный столбец задает веса критериев с точки зрения поставленной цели.

Глубинное интервью проводилось среди владельцев малого и среднего бизнеса: кофейни, рестораны, салоны красоты, фестивали практик, торговые центры. В опросе приняло участие 28 человек. Полученные результаты приведены на следующих диаграммах:

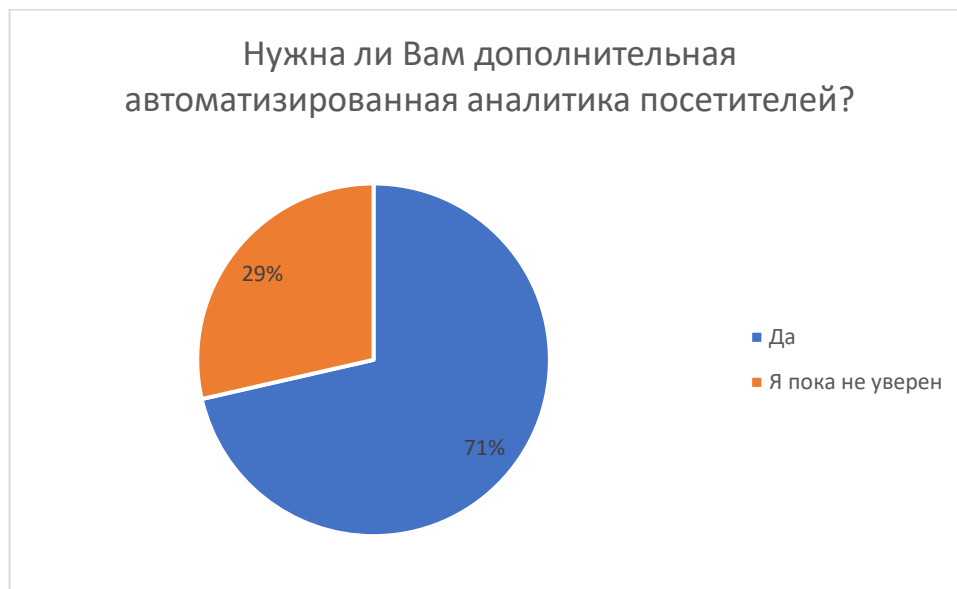


Рис. 1. Круговая диаграмма с результатами ответов на вопрос

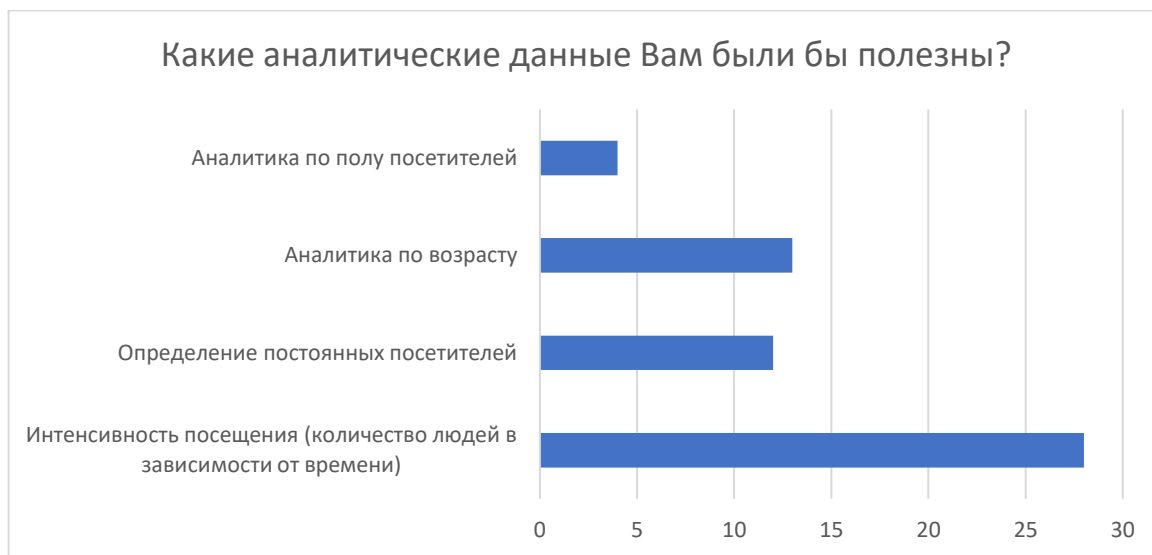


Рис. 2. Столбчатая диаграмма с результатами ответов на вопрос

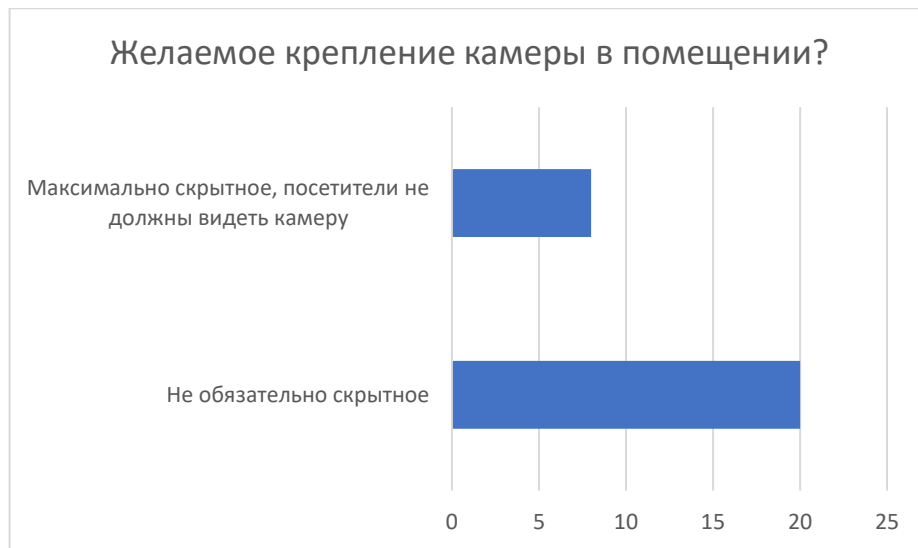


Рис. 3. Столбчатая диаграмма с результатами ответов на вопрос



Рис. 4. Круговая диаграмма с результатами ответов на вопрос

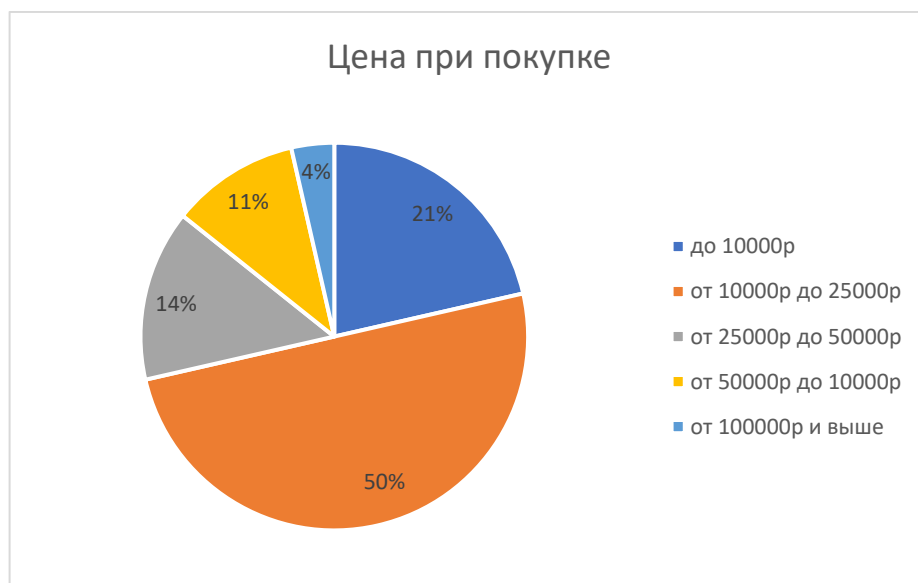


Рис. 5. Круговая диаграмма с результатами ответов на вопрос

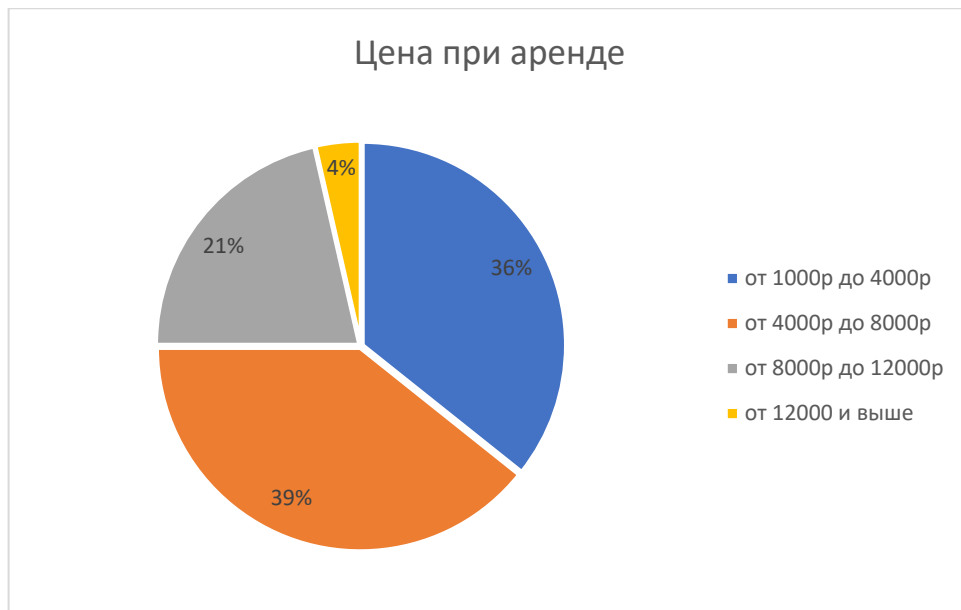


Рис. 6. Круговая диаграмма с результатами ответов на вопрос

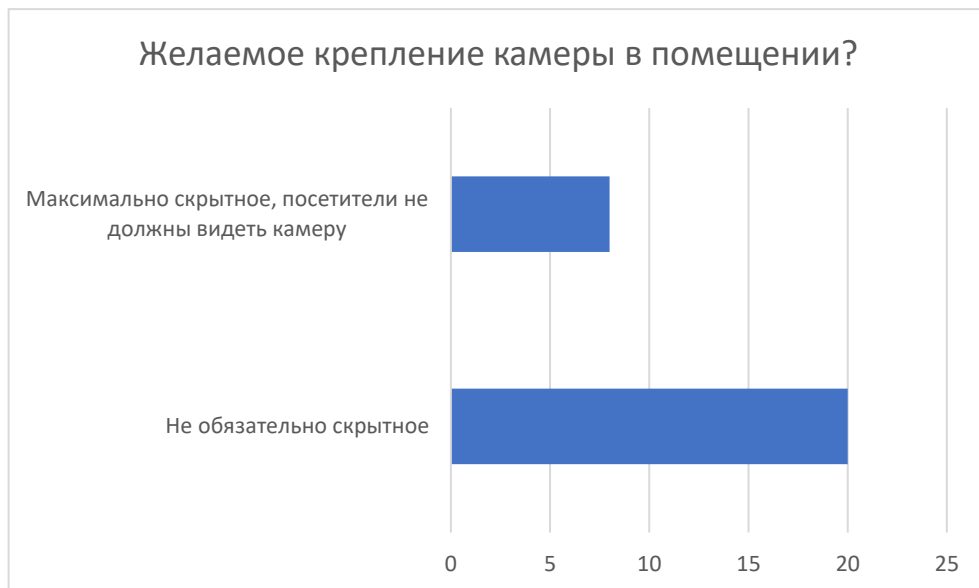


Рис. 7. Столбчатая диаграмма с результатами ответов на вопрос

Можно сделать вывод, что дополнительная аналитика требуется. Наиболее важным параметром аналитики является определение интенсивности посетителей. Камера не обязательно должна быть скрыта. Большая часть опрошенных (57%) хочет арендовать систему и готова потратить на это от 4 до 8 тыс. рублей (39%). На покупку системы большинство (50%) готово потратить от 50 до 100 тыс. рублей.

Для оценки приоритетного способа взаимодействия с системой будем использовать матрицу попарных сравнений:

Таблица 1. Матрица a_{ij}

a_{ij}	Веб-интерфейс	Мобильное приложение	Чат-бот	Email рассылка
Веб-интерфейс	1	3	6	3
Мобильное приложение	1/3	1	2	1
Чат-бот	1/6	1/2	1	1/2
Email рассылка	1/3	1	2	1

Таблица 2. Матрица a_{ij} со строкой суммы

a_{ij}	Веб-интерфейс	Мобильное приложение	Чат-бот	Email рассылка
Веб-интерфейс	1	3	6	3
Мобильное приложение	0,333	1	2	1
Чат-бот	0,167	0,5	1	0,5
Email рассылка	0,333	1	2	1
Сумма	1,833	5,5	11	5,5

Таблица 3. Матрица A_{ij} со столбцом средних значений

A_{ij}	Веб-интерфейс	Мобильное приложение	Чат-бот	Email рассылка	Среднее значение
Веб-интерфейс	0,5456	0,5455	0,5455	0,5455	0,5455
Мобильное приложение	0,1817	0,1818	0,1818	0,1818	0,1818
Чат-бот	0,0911	0,0909	0,0909	0,0909	0,091
Email рассылка	0,1817	0,1818	0,1818	0,1818	0,1818

Таким образом, приоритетным способом взаимодействия с системой является веб-интерфейс.

Для определения приоритетного критерия системы будем использовать матрицу попарных сравнений:

Таблица 4. Матрица a_{ij}

a_{ij}	Цена системы	Надежность устройства	Безопасность	Удобство взаимодействия	Наличие сервисной поддержки	Эргономичность корпуса	Доп. функции аналитики
Цена системы	1	128/132	128/116	128/96	128/128	128/76	128/120

Надежность устройства	132/128	1	132/116	132/96	132/128	132/76	132/120
Безопасность	116/128	116/132	1	116/96	116/128	116/76	116/120
Удобство взаимодействия	96/128	96/132	96/116	1	96/128	96/76	96/120
Наличие сервисной поддержки	128/128	128/132	128/116	128/96	1	128/76	128/120
Эргономичность корпуса	76/128	76/132	76/116	76/96	76/128	1	76/120
Доп. функции аналитики	120/128	120/132	120/116	120/96	120/128	120/76	1

Таблица 5. Матрица a_{ij} со строкой суммы

a_{ij}	Цена системы	Надежность устройства	Безопасность	Удобство взаимодействия	Наличие сервисной поддержки	Эргономичность корпуса	Доп. функции аналитики
Цена системы	1	0,9697	1,1034	1,3333	1	1,6842	1,0667
Надежность устройства	1,0313	1	1,138	1,375	1,0313	1,7368	1,1
Безопасность	0,9063	0,8788	1	1,2083	0,9063	1,5263	0,9667
Удобство взаимодействия	0,75	0,7273	0,8276	1	0,75	1,2632	0,8
Наличие сервисной поддержки	1	0,9697	1,1034	1,3333	1	1,6842	1,0667
Эргономичность корпуса	0,5934	0,5758	0,6552	0,7917	0,5938	1	0,6333
Доп. функции аналитики	0,9375	0,9091	1,0345	1,25	0,9375	1,5789	1
Сумма	6,2185	6,0304	6,8621	8,2916	6,2189	10,4736	6,6334

Таблица 6. Матрица A_{ij} со столбцом средних значений

a_{ij}	Цена	Надежность	Безопасность	Удобство	Сервисная поддержка	Эргономичность	Доп. ф.-ции	Среднее значение
Цена системы	0,1608	0,1608	0,1608	0,1608	0,1608	0,1608	0,1608	0,1608
Надежность устройства	0,1658	0,1658	0,1658	0,1658	0,1658	0,1658	0,1658	0,1658
Безопасность	0,1457	0,1457	0,1457	0,1457	0,1457	0,1457	0,1457	0,1457
Удобство взаимодействия	0,1206	0,1206	0,1206	0,1206	0,1206	0,1206	0,1206	0,1206
Наличие сервисной поддержки	0,1608	0,1608	0,1608	0,1608	0,1608	0,1608	0,1608	0,1608
Эргономичность корпуса	0,0954	0,0955	0,0955	0,0955	0,0955	0,0955	0,0955	0,0955
Доп. функции аналитики	0,1508	0,1508	0,1508	0,1508	0,1508	0,1508	0,1508	0,1508

Таким образом, наиболее приоритетными критериями системы являются надежность устройства, цена и наличие сервисной поддержки.

Результаты экспериментов

По рассмотренным ранее статьям можно сделать вывод, что для непроизводительных устройств на сегодняшний день одним из лучших алгоритмов считается YOLO версии 3. Главной целью при разработке данного алгоритма являлось увеличение скорости работы. Также был создан алгоритм YOLO Tiny версии 3, в котором содержится меньшее количество параметров в модели, что позволило еще больше увеличить скорость предсказания.

В основе алгоритма YOLO лежит принцип деления изображения на ячейки путем наложения сетки. Затем для каждой ячейки модель предсказывает координаты зоны обнаружения и выдает оценку уверенности для этих полей. Затем оценка уверенности для каждой зоны обнаружения умножается на вероятность принадлежности к классу для получения окончательной оценки.

При разработке данной системы на наборе данных SCUT-HEAD была обучена модель YOLO Tiny версии 3, реализованная на фреймворке машинного обучения Tensorflow, который обладает инструментами для переноса обученной модели на устройство. При обучении модели мы обратили внимание на следующие параметры: Intersection Over Union (IOU), anchor boxes и classification score. Для нахождения значений anchor boxes использовался алгоритм Kmeans: с его помощью подбирались ячейки с определенными параметрами ширины-высоты предполагаемого объекта на изображении. Намного сложнее обстояли дела с IOU и classification score. Параметр IOU отвечает за отношение пересечения площадей ячеек и объединение этих площадей. В наборе данных для обучения попадают ситуации, когда одновременно в кадре присутствуют объекты с разной удаленностью от камеры, что усложняет подбор параметров. Ниже будут приведены фотографии с различными версиями IOU и classification score.



Рис. 8. IOU = 0.85, classification score = 0.25

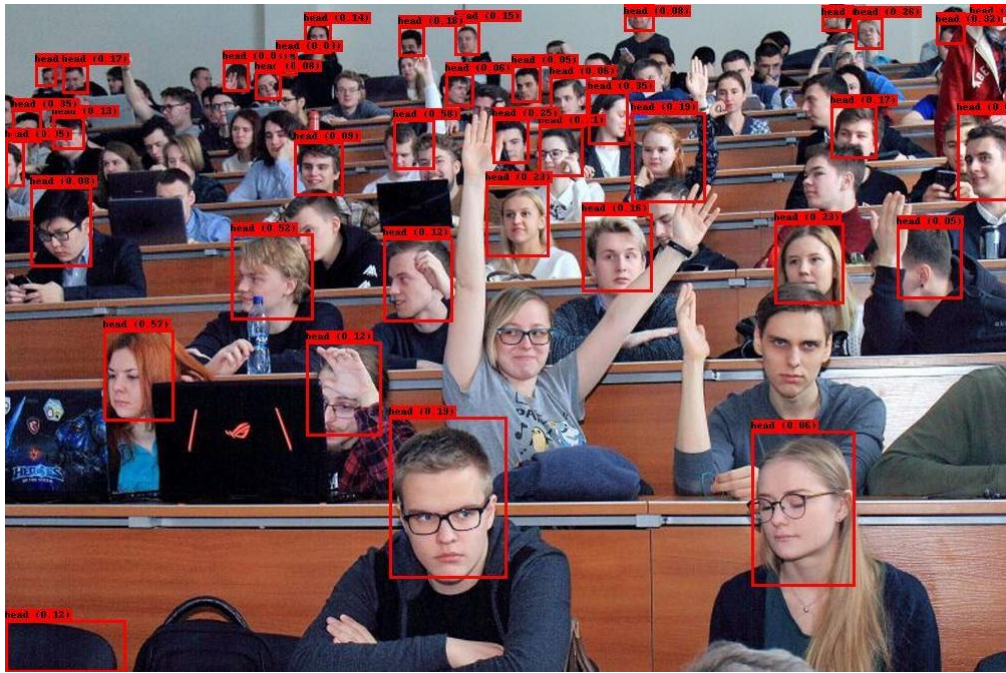


Рис. 9. IOU = 0.1, classification score = 0.05

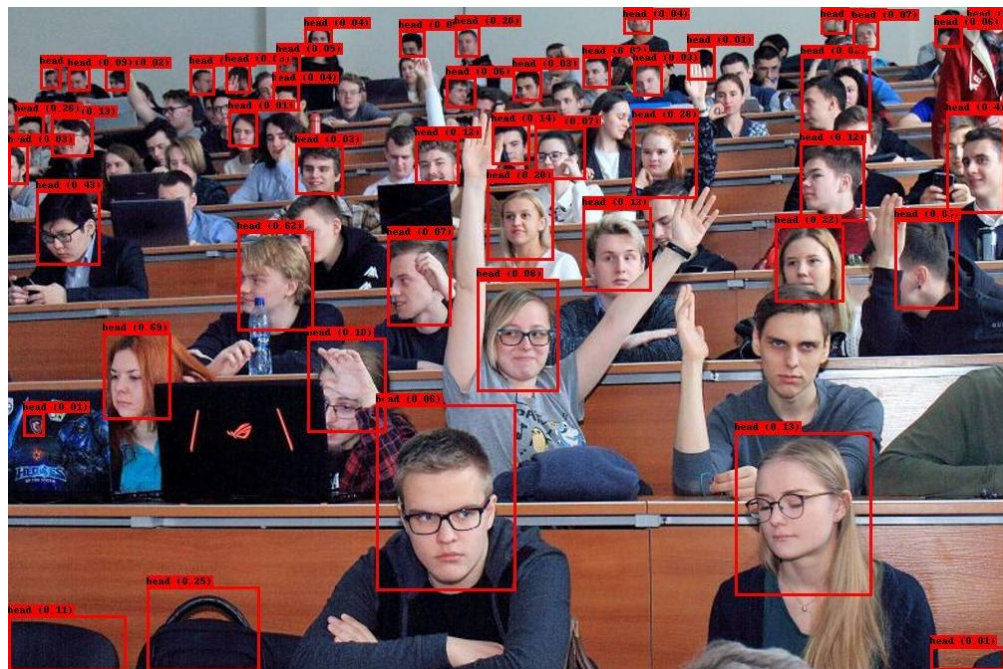


Рис. 10. IOU = 0.01, classification score = 0.01

Данные случаи показывают, что с уменьшением метрик IOU и classification score появляются ложные предсказания, однако одновременно с этим увеличивается и количество правильно найденных голов в сложных ситуациях. Для использования модели в различных ситуациях необходимо адаптировать параметров IOU и classification score.

Полученная модель запускалась для работы на Raspberry Pi 3. Для поиска объектов на изображении размером 640x480 было потрачено 2 секунды. Используя дополнительно Tensorflow Lite, можно получить упрощенную версию модели данной модели. При работе с Tensorflow Lite и Raspberry Pi следует учитывать, что библиотека из

piwheels не будет работать, а значит, следует скомпилировать Tensorflow из исходного кода при помощи Bazel. Использование полученной оптимизированной модели дает прирост по скорости предсказания в 2 раза, один снимок с веб-камеры обрабатывается 1 секунду.

Для реализации интерфейса получения информации пользователем от системы было необходимо создать веб-интерфейс. Для решения подобной задачи для языка Python существуют следующие библиотеки: Flask, Django, Tornado и другие. Однако особенность реализации текущего интерфейса заключается в том, что пользователю необходим удобный просмотр получаемых данных о количестве людей в помещении в течение времени. Для удобства и наглядности используются графики, они помогают быстро оценить ситуацию и понять тренд в данных. Для построения графиков в языке Python существует также большое количество библиотек: Matplotlib, Seaborn, Plotly и другие. При реализации веб-интерфейса для текущего проекта была выбрана библиотека Plotly по следующим причинам: она позволяет создавать интерактивные графики, которые могут быть анимированными и отражать самые последние данные подсчета числа людей в помещении, и интерактивные графики, которые позволяют просматривать историю за каждый предыдущий день. Дополнительное использование инструмента Dash позволяет создать удобный веб-интерфейс для доступа к графикам. Dash использует для создания веб-сервера фреймворк Flask, который в свою очередь не требователен к ресурсам и может работать на Raspberry Pi параллельно с нейронной сетью.

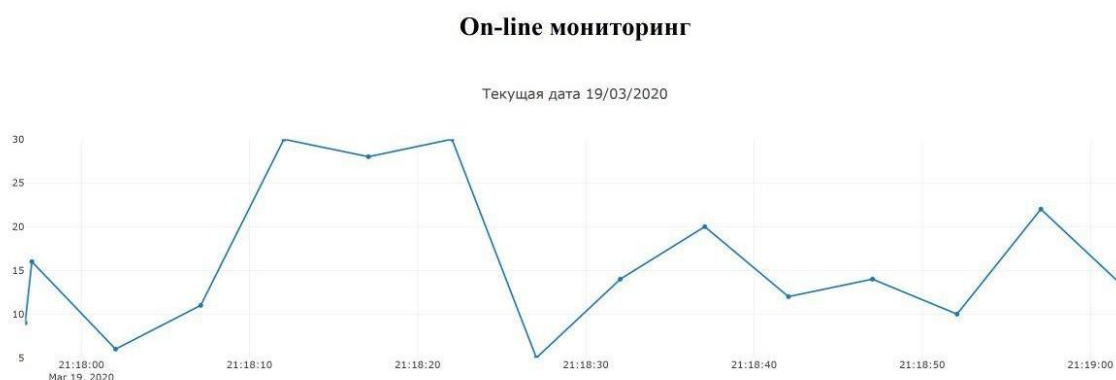


Рис. 11. Веб-интерфейс для on-line мониторинга

Просмотр исторических данных

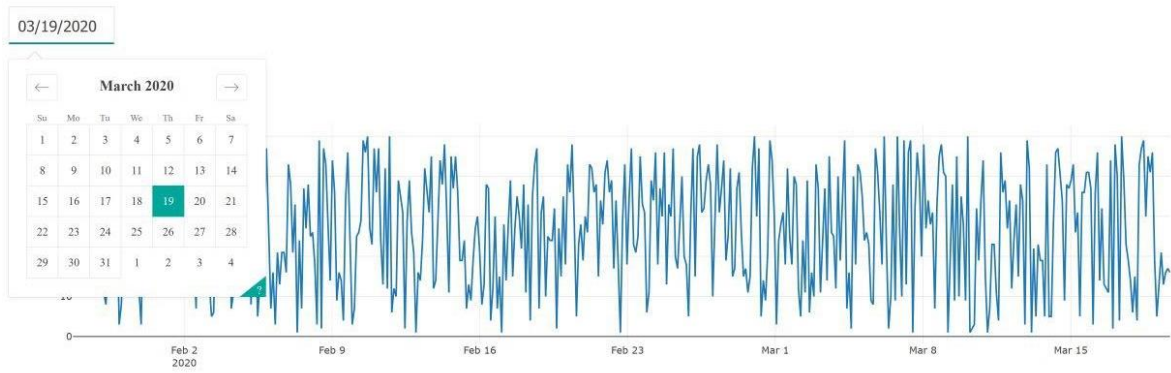


Рис. 12. Веб-интерфейс для просмотра исторических данных



Рис. 13. Фотография прототипа системы «Умная камера». Положение 1



Рис. 14. Фотография прототипа системы «Умная камера». Положение 2

Выводы и дальнейшие перспективы исследования

Созданный прототип системы «Умная камера» распознает в режиме реального времени человеческие головы с изображения, получаемого с камеры, и подсчитывает их количество, выводя результаты в виде графика в веб-интерфейс системы. Модель запускается на Raspberry PI 3. Скорость распознавания с изображения для модели Tensorflow составляет 2 секунды, для Tensorflow Lite – 1 секунду. В дальнейшем планируется расширить набор функций для аналитики: аналитика по возрасту, определение постоянных посетителей и аналитика по полу.

Список литературы

1. Интернет-магазин «VORUNET» : [сайт]. URL: <https://vorunet.ru/schetchiki-posetitelej/schetchik-rstat-real-2d-light/>
2. Track Expert : [сайт]. URL: https://trackexpert.cvizi.com/?utm_source=yandex&utm_medium=cpc&utm_campaign=42827301&utm_content=7741098309&utm_term=%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0%20%D0%BF%D0%BE%D0%B4%D1%81%D1%87%D0%B5%D1%82%D0%B0%20%D0%BF%D0%BE%D1%81%D0%B5%D1%82%D

[0%B8%D1%82%D0%B5%D0%BB%D0%B5%D0%B9&block=premium.4&_openstat=ZGlyZWNOlnlhbmRleC5ydTs0MjgyNzMwMTs3NzQxMDk4MzA5O3lhbmRleC5ydTpwcMvtaXVt&yclid=7620831619647830782](https://www.zencount.ru/?gclid=Cj0KCQiAxfzvBRCZARIsAGA7YMwwxeoNnPhGihNfZZuLLnShHayF63MhAocpIF21mLL3VGXAvziWlmoaAmGtEALw_wcB)

3. ZEN : [сайт]. URL:
https://www.zencount.ru/?gclid=Cj0KCQiAxfzvBRCZARIsAGA7YMwwxeoNnPhGihNfZZuLLnShHayF63MhAocpIF21mLL3VGXAvziWlmoaAmGtEALw_wcB
4. MegaCount : [сайт]. URL: <https://megacount.ru/video#характеристики>
5. Terada K. et al. A counting method of the number of passing people using a stereo camera //IECON'99. Conference Proceedings. 25th Annual Conference of the IEEE Industrial Electronics Society (Cat. No. 99CH37029). – IEEE, 1999. – Т. 3. – С. 1318-1323.
6. Yang D. B. et al. Counting people in crowds with a real-time network of simple image sensors //null. – IEEE, 2003. – С. 122.
7. K. Chen, C. C. Loy, S. Gong, and T. Xiang. Feature mining for localised crowd counting. InBMVC, 2012
8. A. B. Chan, Z. S. Liang, and N. Vasconcelos. Privacy pre-serving crowd monitoring: Counting people without people models or tracking. InCVPR, 2008
9. K. Chen, S. Gong, T. Xiang, Q. Mary, and C. C. Loy. Cumulative attribute space for age and crowd density estimation. InCVPR, 2013.
10. C. C. Loy, S. Gong, and T. Xiang. From semi-supervised to transfer counting of crowds. InICCV, 2013.
11. Zhang C. et al. Cross-scene crowd counting via deep convolutional neural networks //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2015. – С. 833-841.
12. Papageorgiou C. P., Oren M., Poggio T. A general framework for object detection //Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271). – IEEE, 1998. – С. 555-562.
13. Viola P. et al. Robust real-time object detection //International journal of computer vision. – 2001. – Т. 4. – №. 34-47. – С. 4.
14. Redmon J. et al. You only look once: Unified, real-time object detection //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2016. – С. 779-788.
15. Zhang S. et al. Single-shot refinement neural network for object detection //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. – 2018. – С. 4203-4212.
16. Context-aware CNNs for person head detection : [сайт]. URL:
<https://www.di.ens.fr/willow/research/headdetection/>

17. Research Data and the Stanford Digital Repository : [сайт]. URL:
<https://exhibits.stanford.edu/data/catalog/sx925dc9385>
18. GitHub : [сайт]. URL: <https://github.com/HCIILAB/SCUT-HEAD-Dataset-Release>