

Санкт-Петербургский государственный университет

ГАВРИЛОВА Александра Николаевна

Выпускная квалификационная работа

Обнаружение и подсчет лиц на изображении

Направление 02.03.02

«Фундаментальная информатика и информационные технологии»

ООП СВ.5003.2016 «Программирование и информационные технологии»

Научный руководитель:
доцент кафедры КТС,
к.ф.-м.н. Погожев Сергей
Владимирович

Рецензент: доцент кафедр-
ры КММС, к.т.н Гришкин
Валерий Михайлович

Санкт-Петербург
2020

Содержание

Введение.....	3
Постановка задачи.....	4
Обзор литературы.....	5
Глава 1. Описание использованных методов.....	9
1.1. Метод detectMultiScale библиотеки OpenCV.....	9
1.2. Метод библиотеки OpenCV, основанный на нейронных сетях.....	10
Глава 2. Исследование зависимости методов от входных параметров.....	12
2.1. Применение методов к примерам изображений.....	12
2.2. Подготовка базы изображений.....	14
2.3. Проведение исследования.....	14
2.4. Результаты исследования.....	15
Глава 3. Автоматизация подбора параметров.....	23
3.1. Применение автоматизации к трем группам изображений.....	23
3.2. Использование комбинации методов.....	24
3.3. WEB-приложение.....	26
Заключение.....	29
Список литературы.....	30

Введение

Методы обнаружения лиц на изображении имеют широкий спектр применения. Обнаружение лица является первым этапом при выделении признаков для дальнейшей идентификации, при определении пола человека, его этнической принадлежности, при распознавании эмоций. Методы обнаружения лиц применяются при подготовке баз изображений, когда необходимо оставить только те части изображений, на которых присутствуют лица. Кроме того, методы обнаружения лиц могут применяться в задаче подсчета количества людей на изображении. Это может быть необходимо для того, чтобы быстро оценить количество присутствующих людей на конференции, на лекции в университете, на музыкальном концерте, на митинге, в общественном транспорте и при других обстоятельствах, когда присутствует большое количество людей. Создание WEB-приложения, осуществляющего данные функции, позволило бы сделать обнаружение и подсчет лиц на изображении доступными для широкого круга людей.

Постановка задачи

В задаче подсчета лиц на изображении основным этапом является обнаружение лиц. Существует большое количество методов обнаружения лиц на изображении, и важно выбрать метод, подходящий для конкретной задачи. Для данных методов важным критерием качества является не только количество верно определенных лиц, которое необходимо максимизировать, но и количество неверно определенных лиц, которое нужно минимизировать. Добиться одновременной максимизации одного критерия и минимизации другого оказывается невозможным, и требуется найти наилучшее решение, которое было бы допустимым по обоим показателям. При этом для объектов разных размеров подходящее значение входных параметров методов может значительно отличаться. Поэтому следует использовать значения входных параметров методов, подходящие для обнаружения объектов на конкретном изображении.

Применение методов для обнаружения и подсчета лиц в реальных задачах требует написания программ. При этом использование данных методов может быть востребовано среди людей, которые никогда не использовали языки программирования. По этой причине необходимо дать возможность людям, которые не являются специалистами в данной области, использовать методы обнаружения и подсчета лиц на изображении через интерфейс WEB-приложения.

Поставленная задача может быть разделена на следующие этапы:

- 1) изучение существующих методов обнаружения лиц и их реализаций;
- 2) выбор методов и исследование зависимости качества обнаружения от входных параметров;
- 3) написание программы с применением выбранных методов;
- 4) создание WEB-приложения, осуществляющего обнаружение и подсчет лиц на изображении.

Обзор литературы

В настоящее время повсеместно происходит внедрение технологий идентификации и трекинга лиц, важным этапом в применении которых является обнаружение лица на изображении. Поэтому данная область активно развивается, и появляются новые методы. Задача обнаружения лиц является частным случаем задачи обнаружения объектов на изображении. Существуют различные методы обнаружения объектов.

Такие методы, как R-CNN [1], Fast R-CNN [2], Faster R-CNN [3] являются двухэтапными. Первый этап — определение областей, наиболее вероятно содержащих искомый объект, на втором этапе происходит поиск признаков в данных областях, по которым детектор делает предположение о наличии в области объекта, его принадлежности к определенному классу и о расположении ограничивающего данный объект прямоугольника. Нейронная сеть SPP-Net [4] в отличие от R-CNN [1] осуществляет экстракцию признаков только один раз с полного изображения, а не отдельно с каждой из областей, что значительно увеличивает скорость работы. Кроме того, данная нейронная сеть позволяет обрабатывать изображения разных размеров, в отличие от предшествующих сверточных нейронных сетей, обрабатывающих изображения фиксированного размера. В Fast R-CNN [2] используется слой подвыборки, аналогичный используемым слоям подвыборки в SPP-Net [4], но содержащий только один уровень пирамиды. Данный метод позволяет увеличить точность и скорость обнаружения. Faster R-CNN [3] в отличие от Fast R-CNN [2] достигает большей скорости обнаружения за счет другого алгоритма определения областей, наиболее вероятно содержащих искомый объект. Несмотря на это, все эти методы являются недостаточно быстрыми для детектирования большого количества объектов в реальном времени.

Одноэтапные методы имеют большую скорость обработки изображений и позволяют осуществлять обнаружение объектов в реальном

времени. Детектор YOLO [5] позволяет добиться классификации объектов в реальном времени при точности более чем в 2 раза превосходящей появившийся до него детектор 30Hz DPM [6]. В той же статье [5] представлен детектор Fast YOLO, который позволяет добиться в 3 раза большей скорости работы, чем YOLO, при этом по точности обнаружения он в 2 раза превосходит детектор 30Hz DPM [6]. На данный момент известны улучшенные версии детектора YOLO — YOLOv2 [7], YOLOv3 [8]. В статье [5] упоминается, что в задачах обнаружения одного класса объектов, например, лиц, вычисления являются избыточными, так как кроме обнаружения объектов YOLO определяет принадлежность к одному из множества классов, что будет избыточным для данной задачи. Поэтому при применении подобных методов в задачах обнаружения лиц необходимы дополнительные изменения в структуре нейронной сети, чтобы избежать лишних вычислений. Еще одним одноэтапным методом является Single Shot Multibox Detector (SSD) [9]. В SSD используются сверточные слои для последовательного уменьшения размера карты признаков, что позволяет обнаруживать объекты разного размера. Существует легковесная версия детектора на основе MobileNet [10], которая дает возможность использовать SSD на мобильных телефонах, но при этом происходит снижение качества работы детектора. Существуют доступные реализации данных методов, которые распознают большое количество категорий объектов.

Помимо методов, основанных на обучении нейронных сетей, существуют методы, основанные на применении каскада классификаторов. К таким методам относится метод, использующий каскад гистограмм направленных градиентов [11] и метод Viola – Jones [12], в котором используются признаки Хаара.

В методе, основанном на использовании признаков edgelet features [13], человек на изображении рассматривается как совокупность различных частей,

которые ищутся отдельными детекторами и затем объединяются с использованием функции вероятности.

Для подсчета количества людей в толпе используются методы предсказания плотности толпы. Некоторые методы строят регрессию, анализируя изображение целиком [14], но они не учитывают изменчивость плотности на изображении. Поэтому появились методы, которые предполагают разделение изображений на области и построение регрессии плотности толпы на этих областях отдельно [15]. Данные методы не подходят для задач, в которых необходимо получить расположение конкретного лица и ограничивающий прямоугольник вокруг него для продолжения исследования, например, определения пола, эмоций, идентификации человека, так как они не строят ограничивающие прямоугольники, а ставят в соответствие человеку точку на изображении. Методы основаны на плотности толпы в целом, выделение и определение положения для конкретной личности становится невозможным.

Были предложены методы, которые позволяют помимо плотности толпы предсказывать ограничивающие прямоугольники [16, 17]. Преимуществом данных методов является возможность обучения на базах изображений, в аннотации к которым отсутствуют координаты ограничивающих лица прямоугольников, присутствуют только координаты одной точки, соответствующей человеку на изображении, что увеличивает количество баз изображений, подходящих для обучения. При этом детекторы предсказывают возможные ограничивающие прямоугольники.

Существуют различные библиотеки, позволяющие обнаруживать лица на изображении. Согласно технической документации, библиотека Amazon Rekognition [18] позволяет находить лица на изображении, определять расположение глаз, эмоции человека и сравнивать лица с разных изображений, но при обработке рассматриваются только до 100 самых

крупных лиц на изображении. Библиотека Dlib [19] предоставляет методы обнаружения лиц, основанные на нейронных сетях или гистограммах направленных градиентов, однако, в технической документации указано, что методы не предназначены для поиска лиц маленького размера. Поэтому данные библиотеки не рассматриваются в качестве возможных для решения поставленной задачи. Библиотека OpenCV [20] содержит методы для обнаружения лиц на изображении, основанные на алгоритме Viola – Jones [12], а так же модуль DNN (Deep neural networks), позволяющий применять нейронные сети для обнаружения лиц. В отличие от описанных выше библиотек, в технической документации библиотеки OpenCV [20] отсутствуют ограничения на размер и количество лиц на изображении. Метод Viola – Jones [12] является часто применяемым методом обнаружения лиц разного размера, основанным на использовании каскада классификаторов. При использовании модуля DNN (Deep neural networks) возможно использование метода, основанного на Single Shot Multibox Detector (SSD) [9], который позволяет за один этап обработки изображения находить лица разных размеров. Оба метода позволяют получить координаты ограничивающих прямоугольников для найденных лиц. В основе методов лежат отличающиеся друг от друга алгоритмы. Они могут по-разному работать на разных видах изображений, поэтому возможно улучшить качество обнаружения, используя комбинацию методов. OpenCV [20] является широко распространенной среди разработчиков библиотекой с открытым исходным кодом. Кроме методов обнаружения, библиотека предоставляет большое количество различных методов обработки изображений. Поэтому в процессе решения поставленной задачи рассматриваются методы библиотеки OpenCV [20].

Глава 1. Описание использованных методов

1.1. Метод detectMultiScale библиотеки OpenCV

Использовался метод `detectMultiScale` класса `CascadeClassifier` библиотеки OpenCV [20], основанный на алгоритме Viola – Jones [12]. В данном методе обнаружения лиц на изображении используются признаки Хаара, которые представляют собой смежные прямоугольные области. Рассматриваются признаки, состоящие из двух, трех и четырех областей (рис.1). Для признаков из двух областей значение признака вычисляется как разность между суммами интенсивностей пикселей, попавших в разные области. Для признаков из трех областей значение признака – разность между суммами интенсивностей пикселей во внешних областях и суммой интенсивностей в центральной области. Для признаков, состоящих из четырех областей — разность сумм интенсивностей пикселей в диагональных областях. Отбор подходящих признаков при обучении классификатора осуществляется с помощью алгоритма Adaptive boosting [21]. Итоговый классификатор представляет собой каскад классификаторов, каждый из которых осуществляет более сложную проверку на наличие объекта в области изображения. Если классификатор на текущем уровне отвергает наличие объекта, то данная область изображения далее не обрабатывается классификаторами следующих уровней. Таким образом, на начальных этапах алгоритма отбрасывается большое количество областей, что значительно увеличивает скорость работы. Для того чтобы находить объекты разного размера и в разных частях изображения, детектор перемещается по изображению, на каждом шаге размер изображения уменьшается в заданное число раз.

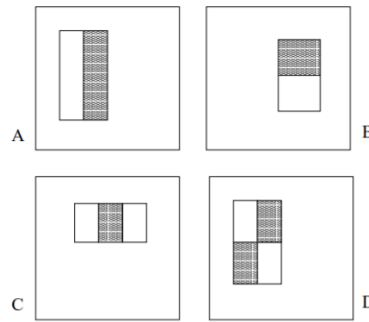


Рис. 1. Признаки Хаара [12]

При работе с детектором поиска лиц на основе признаков Хаара был применен метод `detectMultiScale` библиотеки OpenCV [20]. На вход данного метода подается полутоновое изображение, для его получения был использован метод `cvtColor` библиотеки OpenCV [20]. В качестве результата возвращаются координаты левого верхнего угла, ширина и высота всех найденных ограничивающих объекты прямоугольников. Метод `detectMultiScale` принимает на вход параметр `scaleFactor`, который определяет, во сколько раз уменьшается размер изображения на каждом шаге алгоритма.

1.2. Метод библиотеки OpenCV, основанный на нейронных сетях

Рассматривался модуль DNN (Deep neural networks) библиотеки OpenCV [20]. Обнаружение лиц методами данного модуля осуществлялось с помощью Single Shot Multibox Detector (SSD) [9]. В основе SSD лежит применение сверточных нейронных сетей, при этом в задаче обнаружения лиц рассматриваются только два класса: лицо и задний план. Особенностью данного детектора является последовательное уменьшение размера карты признаков, что позволяет обнаруживать объекты разного размера. Метод `net.forward` возвращает координаты прямоугольных областей, найденных детектором и числовую характеристику, определяющую вероятность присутствия объекта заданного класса в данной прямоугольной области. Далее объект заданного класса считается присутствующим в прямоугольной области, если числовая характеристика превосходит заданный порог. Детектор основан на остаточной нейронной сети ResNet-10. Отличием

остаточных нейронных сетей [22] является наличие между не соседними слоями дополнительных связей, которые представляют из себя тождественные отображения. Такие связи не добавляют дополнительные параметры нейронной сети и не усложняют вычисления.

Глава 2. Исследование зависимости методов от входных параметров

2.1. Применение методов к примерам изображений

Для применения методов на конкретных изображениях была написана программа на языке программирования Python 3.

Для того чтобы увеличить количество верно найденных методом `detectMultiScale` лиц на изображении, использовалось два детектора: для обнаружения лиц в профиль и для обнаружения лиц анфас. Результаты работы детекторов объединялись, и был оставлен только один прямоугольник вокруг найденного лица, даже если оно было обнаружено обоими детекторами. Изначально для объединения результатов использовался следующий подход: если центр найденного детектором для лиц анфас ограничивающего лицо прямоугольника находился внутри прямоугольника, найденного детектором обнаружения лиц в профиль, то далее рассматривался только один из найденных ограничивающих лицо прямоугольников. В результате применения предложенного способа было обнаружено, что он некорректно работал в некоторых случаях. Например, если прямоугольник, верно найденный вторым детектором, находился внутри в несколько раз большего прямоугольника, неверно найденного первым детектором, то вместо верно найденного прямоугольника был оставлен только неверно найденный. Поэтому для дальнейшей работы была использована мера сходства `Intersection over Union`, которая определяется по следующей формуле:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|},$$

где A, B — ограничивающие лица прямоугольники.

Значение порога 0,45 для меры сходства было определено эвристически. Если два ограничивающих прямоугольника имели значение $\text{Intersection over Union}$ превосходящее порог, то считалось, что они ограничивают одно и то же лицо, и дальше рассматривался только один из этих прямоугольников.

При изменении значения параметра `scaleFactor` метода `detectMultiScale` было обнаружено, что значение параметра достаточно сильно влияет на обнаружение объектов разных размеров. Например, параметр `scaleFactor` при значении 1,01 (по умолчанию установлено значение 1,1) хорошо подходит для изображений с большим числом маленьких лиц, но на изображениях с более крупными лицами получается очень большое количество неверно найденных ограничивающих прямоугольников. С другой стороны, увеличение параметра до значения 1,1 дает хорошие результаты на лицах среднего размера, но на изображениях с лицами маленького размера остается большое количество не найденных детектором ограничивающих лица прямоугольников.

Результаты работы метода, основанного на нейронной сети, также зависели от размеров лиц на изображении. На вход по умолчанию подается исходное изображение, размер которого изменяется. При этом, чем меньше становится изображение, тем меньше остается признаков и тем хуже определяются лица маленького размера. Однако, для изображений с крупными лицами возникало большое количество неверно найденных ограничивающих прямоугольников, если изображение не уменьшалось в достаточное число раз.

Поэтому для данных методов было проведено исследование зависимости качества обнаружения лиц разного размера от входных параметров.

2.2. Подготовка базы изображений

В качестве примеров изображений рассматривались изображения из базы «WIDER Face Training Images» [23]. База предоставляет текстовый файл с аннотацией для каждого изображения, в которой указано количество лиц на изображении, координаты левого верхнего угла, ширина и высота ограничивающего прямоугольника для каждого объекта, а также дополнительные параметры, задающие тип изображения, например, освещенность и размытость. Было отобрано 150 изображений и созданы текстовые файлы, содержащие аннотации только к этим изображениям. Изображения были разделены на 3 группы по 50 изображений в каждой, в зависимости от размера максимального по площади ограничивающего объекта прямоугольника, указанного в аннотации. Вычислялась площадь ограничивающего прямоугольника относительно размеров изображения. Осуществлялось вычисление именно относительной площади, а не абсолютной (в пикселах), так как изображения в базе были разного размера. При обработке каждого изображения из соответствующего файла с аннотацией считывалась информация о количестве и расположении ограничивающих объекты прямоугольников. В зависимости от относительной площади наибольшего из прямоугольников, изображение относилось к одной из трех групп: Small Faces (не более 0,002 от площади изображения), Medium Faces (более 0,002 и не более 0,006 от площади изображения), Large Faces (более 0,006 от площади изображения).

2.3. Проведение исследования

Для всех значений параметра к изображению применялся метод обнаружения лиц на изображении и вычислялось количество верно и неверно найденных детектором ограничивающих лицо прямоугольников. Прямоугольник считался определенным верно, если в аннотации к изображению существовал ограничивающий лицо прямоугольник, который

имел с ним значение меры сходства Intersection over Union превосходящее 0,45. Прямоугольник считался определенным неверно, если в аннотации отсутствовал соответствующий ограничивающий лицо прямоугольник.

Для метода detectMultiScale параметр scaleFactor изменялся от 1,01 до 1,2 с шагом 0,005. Изменение входного параметра рассматривалось только в этом промежутке, так как при значении параметра больше 1,2 точность обнаружения была недостаточной даже для больших объектов, а при значении параметра меньше 1,01 происходили ошибки, связанные с недостатком выделенной памяти.

Параметром метода модуля DNN являлось то, во сколько раз будут уменьшены размеры исходного изображения. Параметр изменялся от 0,3 до 1 с шагом 0,1. Нижним значением было выбрано 0,3, так как при меньших значениях линейные размеры подаваемых на вход изображений становились меньше 300, что было недопустимым для работы детектора. Для характеристики, описывающей вероятность присутствия лица в области, был установлен порог 0,5.

2.4. Результаты исследования

В задаче обнаружения лиц на изображении критериями качества являются количество верно найденных лиц, которое необходимо максимизировать и количество неверно найденных прямоугольников, которое необходимо минимизировать. Поэтому в качестве метрик качества были взяты полнота (recall), которая показывает долю верных ответов, и точность (precision), которая зависит от количества неверно найденных прямоугольников. Обе метрики изменяются от 0 до 1. При этом чем выше значения метрик, тем лучше. Вычисления осуществлялись по формулам

$$recall = \frac{TP}{N}, \quad precision = \frac{TP}{TP + FP},$$

где TP — количество верно определенных ограничивающих прямоугольников, FP — количество неверно определенных ограничивающих прямоугольников, N — количество всех ограничивающих объекты прямоугольников в аннотации к изображению.

После этого для каждого значения параметра были вычислены средние по группе значения метрик качества. Значения точности и полноты для метода `detectMultiScale`, соответствующие одному значению параметра представлены на графиках (рис. 2 – 4).

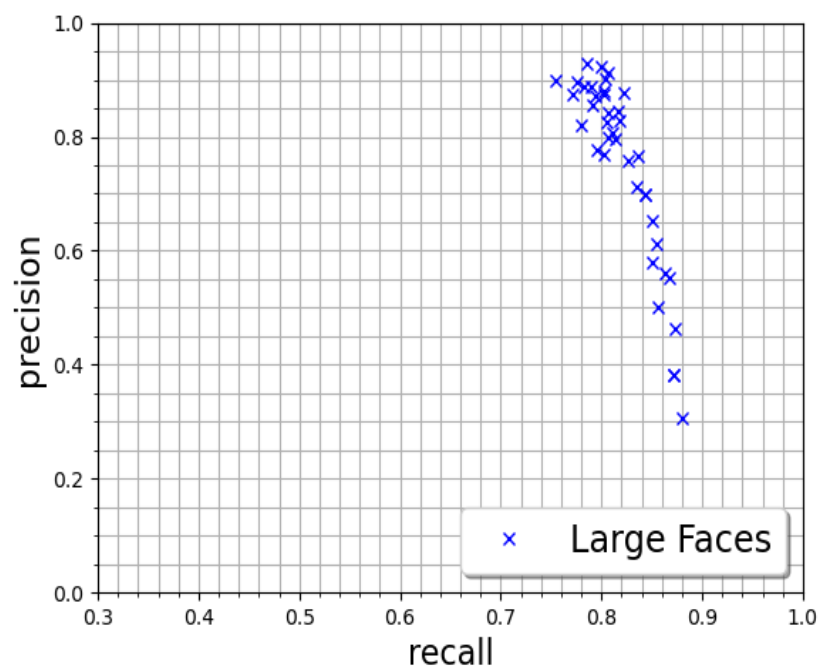


Рис. 2. Значения полноты и точности для группы Large Faces

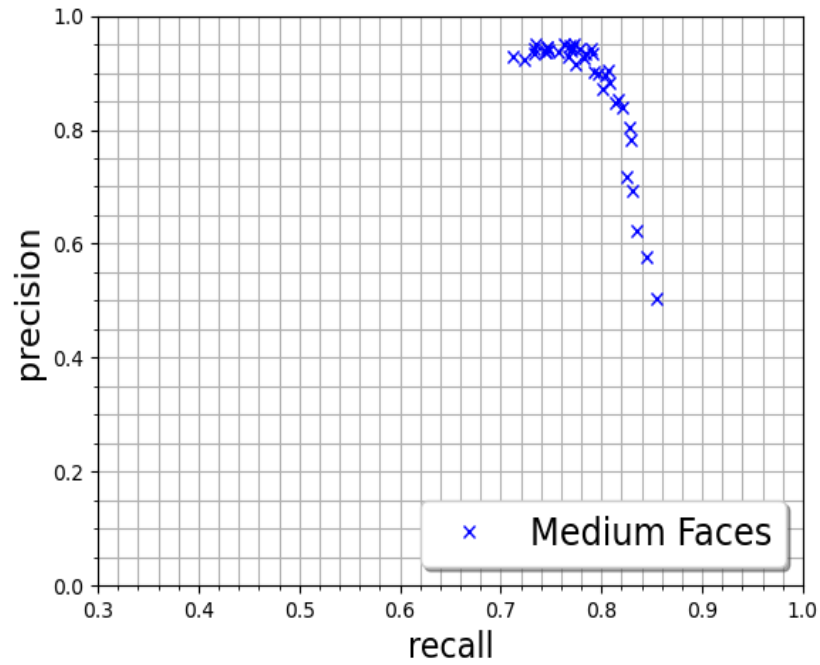


Рис. 3. Значения полноты и точности для группы Medium Faces

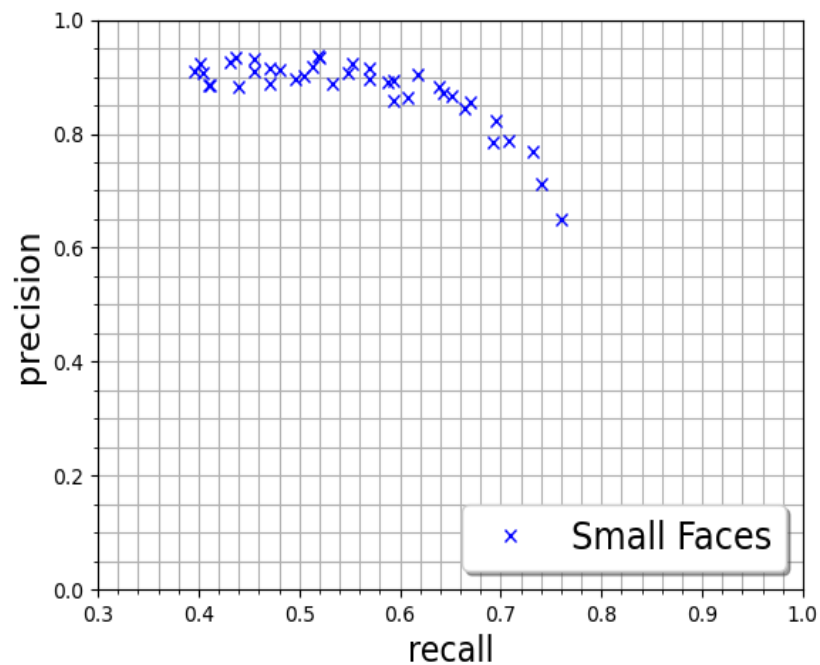


Рис. 4. Значения полноты и точности для группы Small Faces

Из графиков видно, что при увеличении значения полноты, значение точности уменьшается. Для того чтобы определить подходящее значение

параметра, вычислялось гармоническое среднее (F-score) точности и полноты по формуле

$$F = 2 \frac{precision * recall}{precision + recall}$$

Были построены графики зависимости F-score от параметра scaleFactor метода detectMultiScale для каждой из трех групп изображений (рис. 5 – 7).

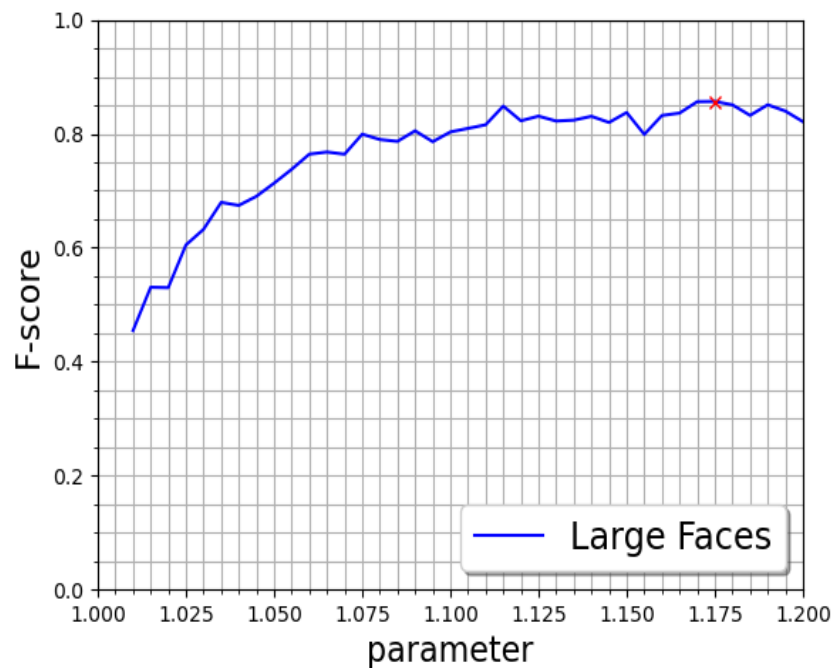


Рис. 5. Группа изображений Large Faces

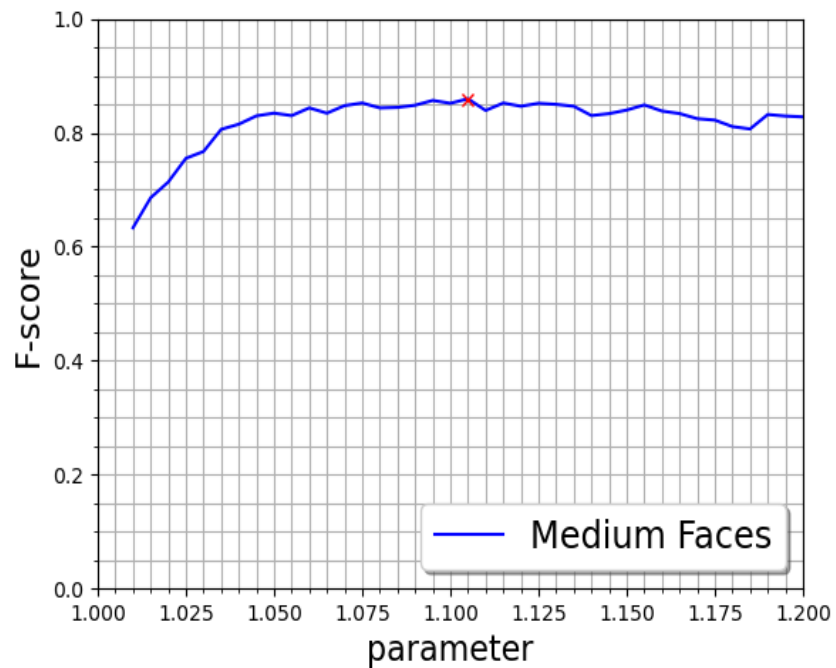


Рис. 6. Группа изображений Medium Faces

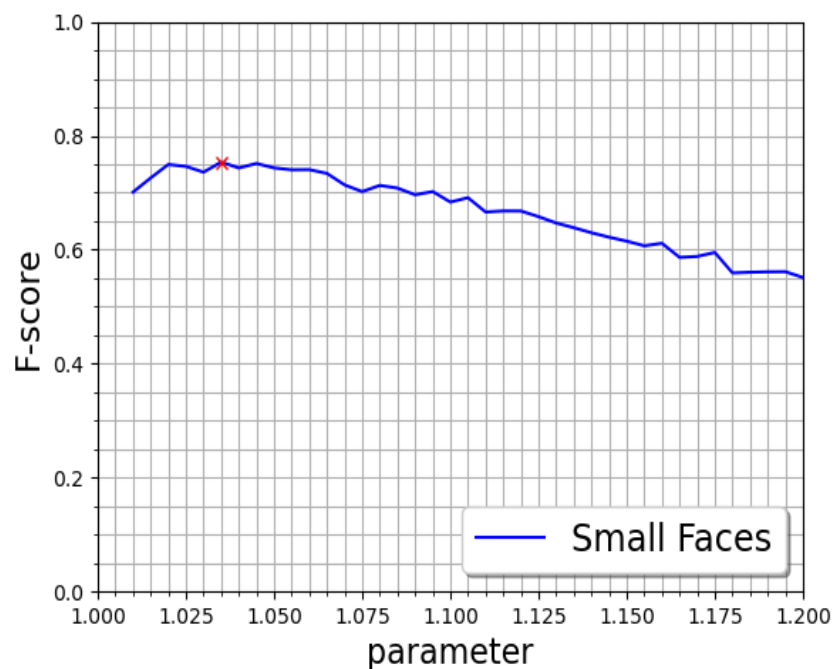


Рис. 7. Группа изображений Small Faces

Для метода, основанного на применении нейронных сетей (SSD) были проведены аналогичные эксперименты и построены графики зависимости F-score от значений входного параметра (рис. 8 – 10).

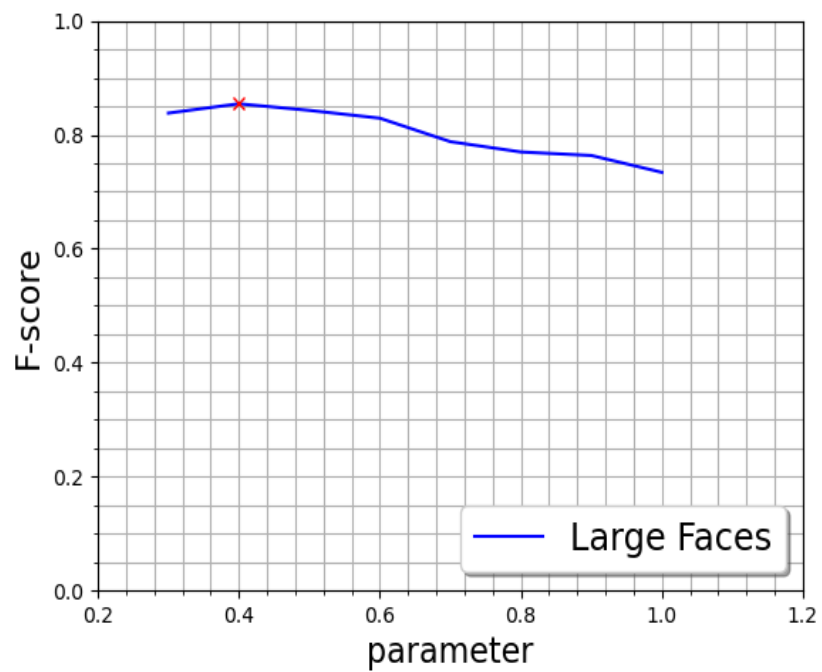


Рис. 8. Группа изображений Large Faces

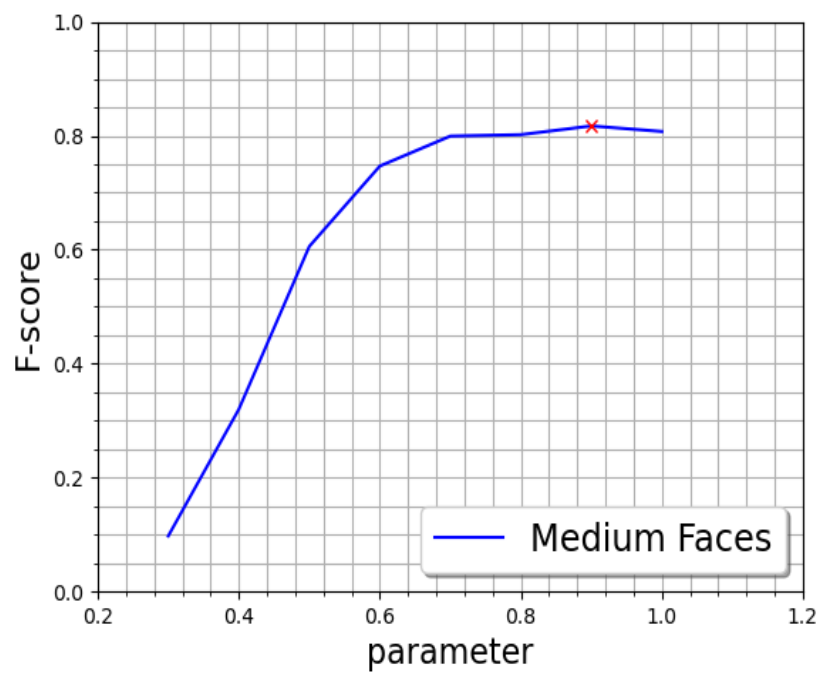


Рис. 9. Группа изображений Medium Faces

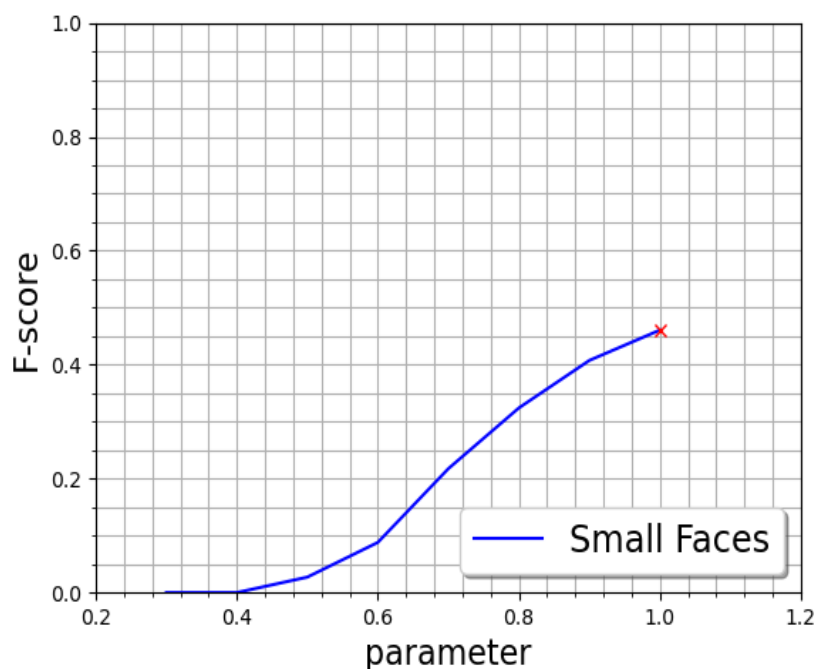


Рис. 10. Группа изображений Small Faces

Кривые ведут себя по-разному для изображений из разных групп. Для каждой группы изображений и для каждого метода было определено значение параметра, при котором F-score принимала наибольшее значение (табл. 1).

Таблица 1. Полученные значения параметра

Группа изображений	Viola – Jones	SSD
Large Faces	1,175	0,4
Medium Faces	1,105	0,9
Small Faces	1,035	1

Однако, как видно из графика (рис. 10), SSD дает плохие результаты на группе изображений с лицами маленького размера. F-score в лучшем случае принимает значение 0,46, в то время как при применении метода detectMultiscale к той же группе изображений лучшее значение F-score равняется 0,75. Это связано с тем, что значение полноты (recall), т.е. доли верно найденных лиц, для SSD на группе с лицами маленького размера

достигает в лучшем случае значения 0,34. Поэтому данный метод не стоит применять при подсчете очень маленьких лиц.

Глава 3. Автоматизация подбора параметров

3.1. Применение автоматизации к трем группам изображений

Для того чтобы автоматизировать подбор параметра, к изображению сначала применялся метод с параметром, соответствующим полученному значению параметра для группы изображений Large Faces, определялась максимальная площадь лица относительно размеров изображения и, если значение было меньше определенного порога, метод применялся повторно со значением параметра, соответствующим изображениям с лицами полученного размера. В качестве начального значения было выбрано значение параметра для группы лиц Large Faces, так как при нем наименьшее количество неверно найденных детектором ограничивающих лицо прямоугольников и наименьшее время обработки изображения детектором. Было проведено тестирование обоих методов на изначальных изображениях и получены результаты F-score для каждой из трех групп изображений. Результаты применения автоматического подбора параметров на тех же изображениях получились не хуже, чем лучшие значения F-score при подборе параметров. По этой причине был сделан вывод, что автоматизация данным образом является приемлемой. Было вычислено время тестирования (табл. 2) и обнаружено, что метод, основанный на нейронных сетях (SSD), работает в несколько раз быстрее метода detectMultiScale (Viola – Jones).

Таблица 2. Время работы методов на трех группах изображений

Группа изображений	Viola – Jones		SSD	
	общее время для 50 изображений, сек.	среднее время обработки изображения, сек.	общее время для 50 изображений, сек.	среднее время обработки изображения, сек.
Large Faces	13,80	0,28	6,2	0,12
Medium Faces	50,07	1,00	15,0	0,30
Small Faces	77,52	1,55	18,4	0,37

3.2. Использование комбинации методов

Так как один из методов работает в несколько раз быстрее другого, но при этом не подходит для обработки лиц очень маленького размера, то было принято решение использовать комбинацию этих методов. Кроме того, метод, основанный на применении нейронных сетей (SSD), менее чувствителен к наклону головы. Метод `detectMultiscale` основан на признаках Хаара, и несмотря на то, что удалось улучшить качество обнаружения лиц при повороте головы добавлением двух детекторов (для обнаружения лиц в профиль и анфас), остается проблема обнаружения лиц при наклоне головы. Применение метода SSD решает данную проблему. На первом этапе для определения максимальной относительной площади ограничивающего лицо прямоугольника на изображении применялся метод SSD, так как обработка изображения этим методом занимает меньшее время. Если вычисленное значение соответствовало группе `Small Faces`, то на втором этапе применялся метод `detectMultiscale`, так как значение полноты на этой группе изображений у метода `detectMultiscale` было значительно выше. Для того, чтобы выбрать, какой метод применять на втором этапе, если размеры лиц на изображении соответствуют группе `Medium Faces`, было проведено тестирование для обоих случаев.

Тестирование было проведено на другой базе изображений — «The Images of Groups» [24]. В данной базе собраны изображения групп людей. Базой предоставляется аннотация, содержащая координаты левого и правого глаза человека, его возраст и пол. Каждое изображение базы было проверено на корректность разметки. Были убраны изображения, размеченные неполностью или ошибочно. Так как базой предоставляются координаты только глаз, а для тестирования необходимы координаты ограничивающих лица прямоугольников, то они были получены автоматически с учетом координат глаз. Затем для каждого изображения вручную была проверена корректность получившихся прямоугольников. Все размеченные изображения оказались подходящими для дальнейшего тестирования.

Оба рассмотренных варианта были протестированы на 2072 изображениях из базы «The Images of Groups» [24]. В первом случае, если вычисленное значение максимальной площади ограничивающего прямоугольника соответствовало группе Medium Faces, то на втором этапе применялся метод detectMultiscale (Viola – Jones). Во втором случае в данной ситуации применялся метод, основанный на нейронных сетях (SSD). Вычислялось количество верно и неверно найденных ограничивающих прямоугольников, аналогично тому, как это было сделано при работе с базой изображений «WIDER Face Training Images» [23]. Вычислялись средние значения точности (precision) и полноты (recall) (табл. 3) и время обработки изображений методами (табл. 4).

Таблица 3. Вычисленные значения метрик качества

Метрика качества	Viola – Jones	SSD
точность (precision)	0,94	0,94
полнота (recall)	0,83	0,82

Таблица 4. Время работы

Метод	Общее время, сек.	Среднее время обработки изображения, сек.
Viola – Jones	418,02	0,20
SSD	386,10	0,19

В результате тестирования было обнаружено, что среднее время обработки изображения детектором SSD на 0,01 меньше времени обработки детектором Viola – Jones. Точность (precision) не зависит от метода, который применяется на данном этапе. Значение метрики полнота (recall) детектора Viola – Jones оказалось больше, чем детектора SSD. При получившейся разнице во времени обработки изображения, предпочтение при выборе лучшего метода отдавалось значениям метрик качества. Поэтому в качестве метода, применяемого на данном этапе был выбран Viola – Jones.

3.3. WEB-приложение

Для использования получившегося алгоритма подсчета лиц на изображении с использованием методов библиотеки OpenCV [20] было написано WEB-приложение. Приложение было написано на языке программирования Python 3 с использованием фреймворка Flask.

В приложении использовался полученный алгоритм. На первом этапе к изображению применялся метод SSD, вычислялось значение относительной площади найденных прямоугольников. В зависимости от максимального значения площади ограничивающего лицо прямоугольника либо возвращался полученный результат, либо параметру присваивалось значение, соответствующее вычисленной максимальной площади, и к изображению применялся метод detectMultiScale с выбранным значением параметра.

Возможно было загрузить изображение на страницу с локального устройства, а также сделать фотографию, используя камеру, и сразу определить на ней расположение и количество лиц.

Результаты отображались на странице WEB-приложения (рис. 11 – 12).

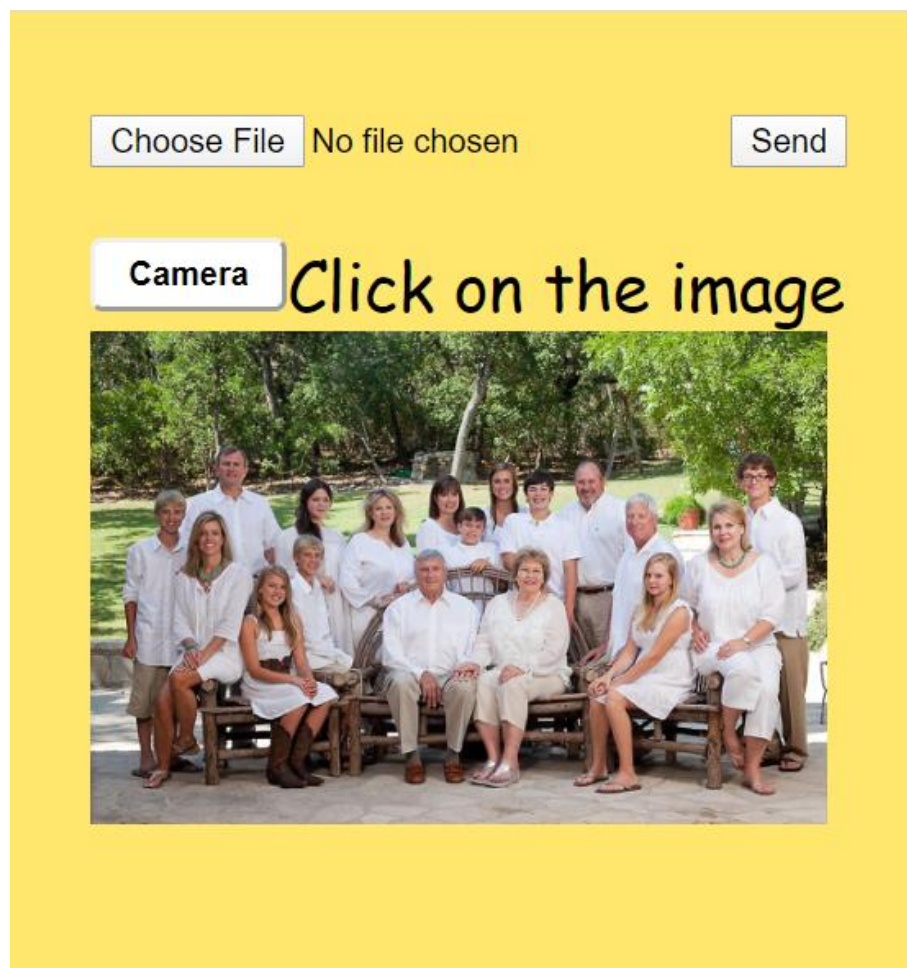


Рис. 11. Загруженное изображение

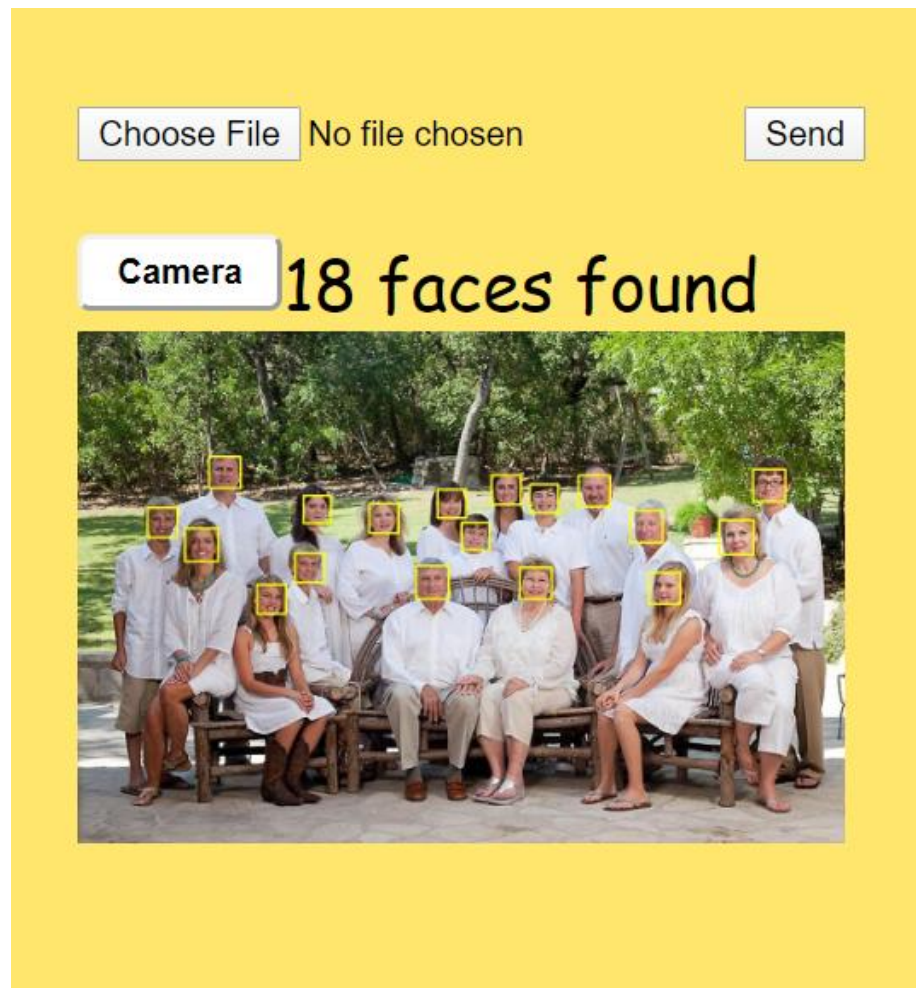


Рис. 12. Результат использования методов

Ссылка на репозиторий с кодом WEB-приложения:

<https://github.com/vsister/Face-detection-and-counting-in-an-image>

Заключение

Методы обнаружения лиц на изображении применяются при решении многих задач. В данной работе представлено применение методов обнаружения лиц для подсчета количества людей на изображении. В результате работы были выполнены все этапы поставленной задачи. Были изучены методы обнаружения лиц на изображении и библиотеки, которые предоставляют их реализации. В качестве используемых методов были выбраны методы библиотеки OpenCV [20]: метод, основанный на алгоритме Viola – Jones [12] и метод, основанный на Single Shot Multibox Detector (SSD) [9]. Для увеличения количества верно найденных лиц первым методом использовалось два детектора: для обнаружения лиц в профиль и для обнаружения лиц анфас, результаты применения которых объединялись с использованием меры сходства Intersection over Union. Проведено исследование зависимости методов библиотеки OpenCV [20] от входных параметров для изображений с лицами разного размера и определены наилучшие значения параметров. Написана программа на языке программирования Python 3 для автоматизации подбора входных параметров и использования комбинации методов для обнаружения и подсчета лиц на изображении. Создано WEB-приложение, позволяющее осуществлять обнаружение и подсчет лиц через интерфейс пользователя.

Список литературы

1. Girshick R., Donahue J., Darrell T., Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation // 2014 IEEE Conference on Computer Vision and Pattern Recognition. 2014. P. 580–587.
2. Girshick R. Fast R-CNN // 2015 IEEE International Conference on Computer Vision (ICCV). 2015. P. 1440–1448.
3. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017. Vol. 39. No. 6. P. 1137–1149.
4. He K., Zhang X., Ren S., Sun J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2015. Vol. 37. No. 9. P. 1904–1916.
5. Redmon J., Divvala S., Girshick R., Farhadi A. You Only Look Once: Unified, Real-Time Object Detection // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017. P. 779–788.
6. Sadeghi M. A., Forsyth D. 30Hz Object Detection with DPM V5 // Computer Vision–ECCV 2014. 2014. P. 65–79.
7. Redmon J., Farhadi A. YOLO9000: Better, Faster, Stronger // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017. P. 6517–6525.
8. Redmon J., Farhadi A. YOLOv3: An Incremental Improvement // ArXiv: 1804.0276. 2018. P. 1–6.
9. Liu W., Anguelov D., Erhan D., Szegedy C., Reed S., Fu C.-Y., Berg A. C. SSD: Single Shot Multibox Detector // European conference on computer vision. 2016. P. 21–37.
10. Howard A. G., Zhu M., Chen B., Kalenichenko D., Wang W., Weyand T., Andreetto M., Adam H. MobileNets: Efficient Convolutional Neural

- Networks for Mobile Vision Applications // arXiv: 1704.04861. 2017. P. 1–9.
11. Zhu Q., Yeh M.-C., Cheng K.-T., Avidan S. Fast Human Detection Using a Cascade of Histograms of Oriented Gradients // 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). 2006. P. 1491-1498.
 12. Viola P., Jones M. J. Robust real-time face detection // International Journal of Computer Vision. 2004. Vol. 57. No 2. P. 137–154.
 13. Wu B., Nevatia R. Detection of multiple, partially occluded humans in a single image by Bayesian combination of edgelet part detectors // Tenth IEEE International Conference on Computer Vision (ICCV'05). 2005. Vol. 1. P. 90-97.
 14. Cho S.-Y., Chow T. W. S., Leung C.-T. A neural-based crowd estimation by hybrid global learning algorithm // IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics). 1999. Vol. 29. No. 4. P. 535–541.
 15. Idrees H., Saleemi I., Seibert C., Shah M. Multi-source Multi-scale Counting in Extremely Dense Crowd Images // 2013 IEEE Conference on Computer Vision and Pattern Recognition. 2013. P. 2547–2554.
 16. Babu Sam D., Peri S. V., Narayanan Sundararaman M., Kamath A., Radhakrishnan V. B. Locate, Size and Count: Accurately Resolving People in Dense Crowds via Detection // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2020.
 17. Liu Y., Shi M., Zhao Q., Wang X. Point in, Box Out: Beyond Counting Persons in Crowds // 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019. P. 6462-6471.
 18. Техническая документация библиотеки Amazon Rekognition [Электронный ресурс]: URL: <https://docs.aws.amazon.com/rekognition/> (дата обращения: 16.05.2020).
 19. Техническая документация библиотеки Dlib [Электронный ресурс]: URL: <http://dlib.net/> (дата обращения: 16.05.2020).

20. Техническая документация библиотеки OpenCV [Электронный ресурс]:
URL: <https://opencv.org/> (дата обращения: 16.05.2020).
21. Freund Y., Schapire R. E.. A decision-theoretic generalization of on-line learning and an application to boosting // Computational Learning Theory: Eurocolt '95. 1995. P. 23–37.
22. He K., Zhang X., Ren S., Sun J. Deep Residual Learning for Image Recognition // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016. P. 770–778.
23. Yang S., Luo P., Loy C. C., Tang X. WIDER FACE: A Face Detection Benchmark // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016. P. 5525–5533.
24. Gallagher A. C., Chen T. Understanding images of groups of people // 2009 IEEE Conference on Computer Vision and Pattern Recognition. 2009. P. 256-263.