даФедеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Южно-Уральский государственный университет»

(национальный исследовательский университет)

Факультет «Математики, механики и компьютерных наук»

Кафедра «Вычислительная механика сплошных сред»

РАБОТА ПРОВЕРЕНА Допустить к Защите

Рецензент, Заведующий кафедрой,

к.ф.-м.н., доцент кафедры МиФА д. ф.-м. н., профессор

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/М.А. Корытова \_\_\_\_\_\_\_\_\_\_\_\_\_\_/Ковалев Ю.М.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2016 г. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2016 г.

**Детектирование ключевых точек на лице человека с использованием сверточных нейронных сетей**

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

ЮУрГУ – 010800.62.2015.838.ВКР

Руководитель

к., ф.-м. н., доцент кафедры

ВМСС

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Н.Л. Клиначева

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2016 г.

Автор работы

студент группы ММиКН-460

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/А.О. Литвинов

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2016 г.

Нормоконтролер – к.ф.-м.н.,

доцент кафедры ВМСС

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/В.К.Рябинин

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2016 г.

Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Южно-Уральский государственный университет»

(национальный исследовательский университет)

Факультет «Математики, механики и компьютерных наук»

Кафедра «Вычислительная механика сплошных сред»

|  |  |
| --- | --- |
|  | УТВЕРЖДАЮ Заведующий кафедрой,  д. ф.-м. н., профессор  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Ю.М. Ковалев  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2016 г. |

**ЗАДАНИЕ**  
студенту группы ММиКН-473

Литвинову Андрею Олеговичу

на выполнение выпускной квалификационной работы

по направлению 011000.62.–Механика. Прикладная математика.

1. **Тема выпускной квалификационной работы** Детектирование ключевых точек на лице человека с использованием сверточных нейронных сетей

**(Утверждена приказом ректора от 27.04.2015 № 838)**

1. **Перечень подлежащих исследованию вопросов**

2.1 Обзор литературы по решению задачи о распространении плоской ударной волны в воздухе, взаимодействии с гетерогенным слоем

2.2 Составление программы расчёта.

2.3 Решение задачи для заданных начальных условий.

1. **Календарный план подготовки выпускной квалификационной работы**

|  |  |  |
| --- | --- | --- |
| Наименование этапов выпускной квалификационной работы | Срок выполнения этапов работы | Отметка о выполнении |
| 1.Обзор литературы | 10.02.16-03.03.16 | выполнено |
| 2. Изучение численного метода для решения поставленной задачи. Решение тестовой задачи. | 04.03.16-25.04.16 | выполнено |
| 3.Получение результатов, формулировка выводов, структурирование текста | 25.04.16-15.05.16 | выполнено |
| 4.Подготовка текста | 16.05.15-01.06.15 | выполнено |
| 5. Проверка и рецензирование работы руководителем, исправление замечаний. Подготовка доклада и текста выступления. Внешнее рецензирование. | 02.06.15-13.06.15 | выполнено |
| 6. Защита выпускной квалификационной работы. | 22.06.16 |  |

**4. Дата выдачи задания 10 февраля 2016 г.**

**Руководитель работы**

к.ф.-м.н., доцент кафедры ВМСС \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Н.Л. Клиначева

Задание принял к исполнению**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/**А.О. Литвинов

УДК 533.6.011.1

**Литвинов А.О.**

Построение математической модели многокомпонентной смеси с учетом сжимаемости частиц инвариантной к преобразованиям Галилея. Литвинов А.О. – Челябинск, 2016. – 35с.

В дипломной работе реализован модифицированный метод крупных частиц для расчета течений в сложных задачах газодинамики. Исследовано влияние частиц на параметры УВ, падающей на гетерогенный слой. Проведен анализ математической модели, предложенной П.Б. Вайнштейном, на инвариантность относительно преобразований Галилея.

В приложении приведен текст программы на языке python

Список лит. – 20 назв.

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 7](#_Toc452556244)

[ПОСТАНОВКА ЗАДАЧИ 8](#_Toc452556245)

[АНАЛЬТЕРНАТИВНЫЕ МЕТОДЫ РЕШЕНИЯ 9](#_Toc452556246)

[Метод главных компонент 9](#_Toc452556247)

[Линейный дискриминантный анализ 10](#_Toc452556248)

[Синтез объектов линейных классов 11](#_Toc452556249)

[Гибкие контурные модели 12](#_Toc452556250)

[Сравнение эластичных графов 12](#_Toc452556251)

[Методы, основанные на геометрических характеристиках лица 14](#_Toc452556252)

[Сравнение шаблонов. 15](#_Toc452556253)

[Скрытые Марковские модели. 16](#_Toc452556254)

[Многослойные нейронные сети. 19](#_Toc452556255)

[Сети Габоровых вейвлетов (GWN). 22](#_Toc452556256)

[Нейронные сети Хопфилда. 22](#_Toc452556257)

[ДЕТЕКТИРОВАНИЕ ЧЕЛОВЕЧЕСКОГО ЛИЦА НА ИЗОБРАЖЕНИИ ДЛЯ ВХОДА СВЕРТОЧНОЙ СЕТИ 24](#_Toc452556258)

[Интегральное представление изображений 24](#_Toc452556259)

[Хаар-подобные характеристики 25](#_Toc452556260)

[Метод построения классификатора на основе алгоритма бустинга 26](#_Toc452556261)

[Метод комбинирования классификаторов в каскадную структуру 27](#_Toc452556262)

[НАХОЖДЕНИЕ ТОЧЕК ЛИЦА ПО ДЕТЕКТИРОВАННОМУ СЕКТОРУ 28](#_Toc452556263)

[СТРУКТУРА СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ 28](#_Toc452556264)

[ПРИНЦИП РАБОТЫ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ 30](#_Toc452556265)

[Convolution слой 30](#_Toc452556266)

[Pooling слой 30](#_Toc452556267)

[Fully-connected 31](#_Toc452556268)

[ПРИНЦИП ОБУЧЕНИЯ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ 32](#_Toc452556269)

[Вычисление ошибки 32](#_Toc452556270)

[Вычисление градиента 33](#_Toc452556271)

[АРХИТЕКТУРА СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ В ДАННОЙ РАБОТЕ 34](#_Toc452556272)

[МЕТОДЫ ПРИМЕНЯЕМЫЕ ПРИ ОБУЧЕНИИ 35](#_Toc452556273)

[Обучение на GPU 35](#_Toc452556274)

[Увеличение тестовой выборки 36](#_Toc452556275)

[Предотвращение переобучения 36](#_Toc452556276)

[Dropout 37](#_Toc452556277)

[Локальные минимумы 37](#_Toc452556278)

[Перемешивание обучающий примеров 37](#_Toc452556279)

[Паралич сети 38](#_Toc452556280)

[РЕЗУЛЬТАТЫ 39](#_Toc452556281)

[ЗАКЛЮЧЕНИЕ 40](#_Toc452556282)

[ЛИТЕРАТУРА 41](#_Toc452556283)

[ПРИЛОЖЕНИЕ 42](#_Toc452556284)

[Текст программы на языке python 42](#_Toc452556285)

ВВЕДЕНИЕ

Искусственные нейронные сети в настоящее время получили широкое распространение в различных сферах человеческой деятельности, таких как распознавание и классификация образов, прогнозирование, управление. Пример конкретных задач – автономная машина, распознавание изображений, речи, текстовые чат боты. Это произошло благодаря тому, что они имеют возможность синтезировать какие-либо алгоритмы, вычленяя из больших выборок данных произвольные закономерности. Изначально они построены по примеру биологических нейронных сетей – сетей нервных клеток животного. Один из видов искусственных нейронных сетей - сверточные нейронные сети, в основном применяющиеся для работы с изображениями. Причину этого можно увидеть в предпосылках их зарождения. Сверточные сети построены по примеру обработки визуальной информации у живых существ – в мозге есть области, выделяющие какие-либо инвариантные закономерности в изображениях – например линии.

Целью выпускной квалификационной работы является исследование возможностей применения сверточных нейронных сетей в области детекции произвольных объектов на RGB изображении

Для достижения поставленной цели необходимо исследовать предметную область, исследовать методы для решения аналогичных задач, исследовать вспомогательные методы.

Также требуется разработать архитектуру нейронной сети для конкретной работы – для детекции ключевых точек лица человека. Этот тип задачи относиться к классу регрессионных задач, то есть восстановление каких-либо зависимостей. Аналогичные методы решения возможно применять к схожим проблемам – распознавание рукописного текста, номеров автомобилей, дорожных знаков.

ПОСТАНОВКА ЗАДАЧИ

Данные получаемые с видеокамеры(кадр) в момент времени обозначим . Каждый кадр есть матрица размерности , каждый элемент принадлежит множеству , где – ширина кадра, - высота кадра, R,G, B – множества в диапазоне от 0 до 255, обозначающие кодировки цветов в формате RGB. Видеопоследовательность есть упорядоченная последовательность кадров (), где – количество кадров в видеопоследовательности.

Сформулируем задачу детектирования точек на лице человека. По имеющемуся кадру предсказать положение точек лица человека, предполагая при этом, что в кадре находится только один человека, на расстоянии от 0.5 до 2 метров от камеры. Требуется предсказать положение 8 точек:

* Центр подбородка
* Левый край губ
* Правый край губ
* Центр носа
* Левую височную область
* Левый край левого глаза
* Правую височную область
* Правый край правого глаза

Процесс детекции происходит в два этапа:

1. Детектирование человеческих лиц на изображение, для последующей передачи выделенных областей сверточной нейронной сети, с помощью каскада Хаара
2. Детектирование на выделенной области точек лица с помощью сверточной нейронной сети.

В работе необходимо из видеопоследовательности в режиме реального времени выделять точки лица. В качестве исходных данных выступает видеопоследовательность, полученная с неподвижной камеры, с разрешением пикселей в формате RGB глубиной 8 бит на один канал и частотой кадров 30 в секунду.

Заданные входные данные накладывают на возможные методы решения ограничения в плане оптимизации

АНАЛЬТЕРНАТИВНЫЕ МЕТОДЫ РЕШЕНИЯ

Метод главных компонент

Метод главных компонент (Principal Component Analysis, {PCA}) применяется для сжатия информации без существенных потерь информативности. Он состоит в линейном ортогональном преобразовании входного вектора размерности в выходной вектор размерности , . При этом компоненты вектора являются некоррелированными и, следовательно, общая дисперсия после преобразования остается неизменной. Матрица состоит из всех примеров изображений обучающего набора. Решив уравнение , получаем матрицу собственных векторов , где - ковариационная матрица для , а - диагональная матрица собственных чисел. Выбрав из подматрицу , соответствующую наибольшим собственным числам, получим, что преобразование , где - нормализованный вектор с нулевым математическим ожиданием, характеризует большую часть общей дисперсии и отражает наиболее существенные изменения . Выбор первых главных компонент разбивает векторное пространство на главное (собственное) пространство , содержащее главные компоненты, и его ортогональное дополнение . В качестве индикаторов принадлежности в методе главных компонент используют:

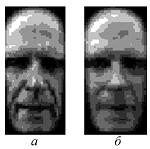
DIFS - distance in feature space, расстояние от образа анализируемого изображения в собственном пространстве, до эталонного образа;

DFFS - distance from feature space, расстояние от представления анализируемого изображения в пространстве наблюдения до проекции эталона в собственном пространстве.



Пример изображений собственных векторов (собственные лица)

Применение для задачи распознавания человека по изображению лица имеет следующий вид. Входные векторы представляют собой отцентрированные и приведенные к единому масштабу изображения лиц. Собственные векторы, вычисленные для всего набора изображений лиц, называются собственными лицами (eigenfaces). Метод главных компонент в применении к изображениям лиц также называют методом собственных лиц (рис. 1). С помощью вычисленных ранее матриц входное изображение разлагается на набор линейных коэффициентов, называемых главными компонентами. Сумма первых главных компонент, умноженных на соответствующие собственные векторы, является аппроксимацией изображения порядка (рис. 2).



Нормализованное изображение лица (*а*) и его реконструкция по 85 главным компонентам (*б*)

Для каждого изображения лица вычисляются его главные компоненты. Обычно берется от 5 до 200 главных компонент. Остальные компоненты кодируют мелкие различия между лицами и шум. Процесс распознавания заключается в сравнении главных компонент неизвестного изображения с компонентами всех остальных изображений. Для этого обычно применяют какую-либо метрику (простейший случай - Евклидово расстояние). Дополнительное повышение надежности достигается за счет дополнительного применения анализа главных компонент к отдельным участкам лица таким, как глаза, нос, рот.

Также метод главных компонент применяется для обнаружения лица на изображении. Для лиц значения компонент в собственном пространстве имеют большие значения, а в дополнении собственного пространства - близки к нулю. По этому факту можно обнаружить, является ли входное изображение лицом. Для этого проверяется величина ошибки реконструкции; чем больше ошибка, тем больше вероятность, что это не лицо. При наличии в наборе изображений лиц вариаций таких, как раса, пол, эмоции, освещение, будут появляться компоненты, величина которых в основном определяется этими факторами. Поэтому по значениям соответствующих главных компонент можно определить, например, расу или пол человека. Основные недостатки PCA таковы. Метод собственных лиц требует для своего применения идеализированных условий таких, как единые параметры освещенности, нейтральное выражение лица, отсутствие помех вроде очков и бород. При несоблюдении этих условий главные компоненты не будут отражать межклассовые вариации. Например, при различных условиях освещенности метод собственных лиц практически неприменим, поскольку первые главные компоненты преимущественно отражают изменения освещения, и сравнение выдает изображения, имеющие похожий уровень освещенности.

Вычисление набора собственных векторов отличается высокой трудоемкостью. Один из способов - это свертка изображений по строкам и столбцам; в такой форме представление изображения имеет на порядок меньший размер, вычисления и распознавание происходит быстрее, но восстановить исходное изображение уже невозможно.

Линейный дискриминантный анализ

Линейный дискриминантный анализ (линейный дискриминант Фишера Linear Discriminant Analysis, {LDA}) использует такую проекцию пространства изображений на пространство признаков, которая минимизирует внутриклассовое и максимизирует межклассовое расстояние в пространстве признаков. В этих методах предполагается, что классы линейно разделимы.

Матрица для проецирования пространства изображения на пространство признаков выбирается из следующего условия:

Где - матрица межклассовой дисперсии, - матрица внутриклассовой дисперсии.

Может существовать до векторов составляющих базис пространства признаков, где - общее число классов. С помощью этих векторов пространство изображений переводится в пространство признаков.

Поскольку работа непосредственно с матрицей затруднительна из-за ее размерности, используется предварительное уменьшение размерности с помощью метода главных компонент, и затем вычисления производятся в пространстве меньшей размерности:

Где - матрица для проецирования в пространство меньшей размерности (пространство главных компонент).

Обычно тренировочный набор содержит изображения лиц при нескольких базовых условиях освещенности, на основе которых при помощи линейных комбинаций можно получить любые другие условия освещенности. Этот метод дает высокую точность распознавания (около 96{\%}) для широкого диапазона условий освещенности, различных выражений лица и наличия или отсутствия очков. Однако остаются невыясненными вопросы, применим ли этот метод для поиска в больших базах данных, может ли метод работать, когда в тренировочной выборке для некоторых лиц имеется изображение только в одних условиях освещенности. Для задачи детектирования лица с помощью LDA-классы лиц и "не лиц" разбивают на подклассы. Вышеописанный метод основывается на предположении о линейной разделимости классов в пространстве изображений. В общем случае такое предположение несправедливо. Инструмент для построения сложных разделяющих поверхностей предлагают нейросетевые методы.

Синтез объектов линейных классов

Данный метод позволяет синтезировать новые изображения объекта (и в частности, изображения лица) для разных ракурсов. Имеется тренировочный набор изображений лиц и только одно изображение нового объекта в определенном ракурсе. Тренировочный набор состоит из изображений объектов того же класса (класс лиц в данном случае), что и новый объект, и включает в себя изображения различных лиц, причем для каждого лица имеются его изображения в широком диапазоне ракурсов. Для нового объекта, имеющего изображение в ракурсе , осуществляется линейное разложение на изображения объектов из тренировочного набора в том же ракурсе, с вычислением коэффициентов , где - количество объектов в тренировочном наборе. Синтез изображения в новом ракурсе для нового объекта осуществляется сложением изображений из тренировочного набора в ракурсе с теми же коэффициентами: . Таким образом, метод позволяет синтезировать изображения нового объекта в различных ракурсах по изображению в одном ракурсе без привлечения сложных трехмерных моделей. Данный метод является перспективным для синтеза изображений в новых ракурсах без привлечения сложных трехмерных моделей, однако вопрос о качестве и количестве примеров в тренировочном наборе остается открытым.

Гибкие контурные модели

В данных методах распознавание производится на основе сравнения контуров лица. Контуры обычно извлекаются для линий головы, ушей, губ, носа, бровей и глаз. Контуры представлены ключевыми позициями, между которыми положение точек, принадлежащих контуру, вычисляются интерполированием. Для локализации контуров в различных методах используется как априорная информация, так и информация, полученная в результате анализа тренировочного набора. Обычно ключевые точки размещаются вручную на наборе тренировочных изображений. При поиске контуров нового лица используется метод симуляции отжига с целевой функцией из двух составляющих. Для первой из них ищется максимум при соответствии интенсивностей пикселов, извлеченных на перпендикулярной контуру линии, аналогичным пикселам из тренировочной выборки, для второй - при совпадении контура с формой контуров тренировочных примеров. Таким образом, извлекается контур черт лица. Для сравнения изображений используются значения главных компонент, вычисленные на наборе векторов, представляющих собой координаты ключевых точек. Главной задачей при распознавании по контурам является правильное выделение этих контуров. В общем виде эта задача по сложности сравнима непосредственно с распознаванием изображений.

Сравнение эластичных графов

В этом методе (Elastic Bunch Graph Matching) лицо представляется в виде графа, вершины которого расположены на ключевых точках лица таких, как контуры головы, губ, носы, и их крайних точках. Каждая грань помечена расстояниями между ее вершинами. В каждой такой точке вычисляются коэффициенты разложения по функциям Габора для пяти различных частот и восьми ориентаций. Набор таких коэффициентов называется *джетом* (jet). Джеты характеризуют локальные области изображений и служат для двух целей: во-первых, для нахождения точек соответствия в заданной области на двух различных изображениях; во-вторых - для сравнения двух соответствующих областей различных изображений. Каждый коэффициент для точек из одной области различных изображений характеризуется амплитудой , которая медленно меняется с изменением положения точки, и фазой , которая вращается со скоростью, пропорциональной частоте волнового вектора базисной функции. Поэтому в простейшем случае для поиска на новом изображении точки с аналогичными характеристиками в функции подобия фазу не учитывают:

Функция подобия с одним джетом в фиксированной позиции и другим с переменной позицией является достаточно гладкой, для того чтобы получить быструю и надежную сходимость при поиске с применением простейших методов таких, как диффузия или градиентный спуск. Более совершенные функции подобия привлекают информацию о фазе. Для различных ракурсов соответствующие ключевые точки отмечаются вручную на тренировочном наборе. Кроме того, чтобы для одного и того же лица представить различные вариации его изображения в одном и том же графе, для каждой точки используются несколько джетов, каждый из которых может соответствовать различным локальным характеристикам данной точки, например, открытому и закрытому глазу. Процесс распознавания неизвестного лица состоит в сравнении графа изображения лица со всеми остальными графами из набора при помощи функции подобия

Левая сумма характеризует подобие джетов, вычисленное с применением фазочувствительной функции, правая - топографическое соответствие, которое пропорционально квадрату разности расстояний между соответствующими вершинами сравниваемых изображений, - количество вершин, - количество граней, - коэффициент относительной важности топографической информации.

В представленном выше виде метод способен достаточно надежно распознавать при изменениях ракурса до 20 ; при больших углах точность распознавания резко уменьшается, функция подобия оказывается более чувствительной к ракурсу, чем к межклассовым различиям. Дальнейшее развитие метода заключается в извлечении коэффициентов важности на основе анализа обучающей выборки. Для каждого джета симплекс-методом вычисляется коэффициент важности, который затем используется в функции подобия. Коэффициенты важности вычисляются из условия максимизации функции подобия для одного и того же лица и минимизации - для различных лиц. Существуют также более ранние разновидности этого метода, которые не используют изначально определенные ключевые точки и структуры графа. Одни из них используют для сравнения решетки джетов, наложенные на изображение, рис. 3. В неизвестном изображении отыскиваются точки соответствия, и затем по найденным точкам строится искаженная решетка и измеряется мера ее искажения для определения наиболее похожего изображения. В других методах точки извлечения джетов изначально образуют решетку, а затем наименее пригодные для распознавания точки отсеиваются в процессе обучения.



Наложенная на изображение эластичная решетка и ее искаженная версия

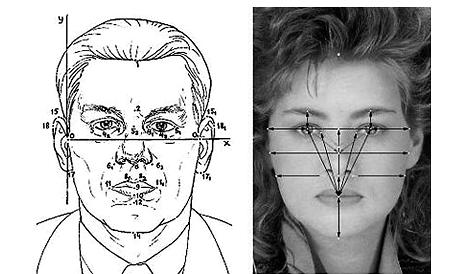
Методы, основанные на геометрических характеристиках лица

Один из самых первых методов - это анализ геометрических характеристик лица. Изначально он применялся в криминалистике и был там детально разработан. Потом появились компьютерные реализации этого метода. Суть его заключается в выделении набора ключевых точек (или областей) лица и последующем выделении набора признаков. Каждый признак является либо расстоянием между ключевыми точками, либо отношением таких расстояний. В отличие от метода сравнения эластичных графов, здесь расстояния выбираются не как дуги графов. Наборы наиболее информативных признаков выделяются экспериментально.

Ключевыми точками могут быть уголки глаз, губ, кончик носа, центр глаза и т. п. рис. 4. В качестве ключевых областей могут служить прямоугольные области, включающие в себя: глаза, нос, рот.

В процессе распознавания сравниваются признаки неизвестного лица с признаками, хранящимися в базе. Задача нахождения ключевых точек приближается к трудоемкости непосредственно распознавания, и правильное нахождение ключевых точек на изображении во многом определяет успех распознавания. Поэтому изображение лица человека должно быть без помех, мешающих процессу поиска ключевых точек. К таким помехам относят очки, бороды, украшения, элементы прически и макияжа. Освещение желательно равномерное и одинаковое для всех изображений. Кроме того, изображение лица должно иметь фронтальный ракурс, возможно, с небольшими отклонениями. Выражение лица должно быть нейтральным. Это связано с тем, что в большинстве методов нет модели учета таких изменений.

Таким образом, данный метод предъявляет достаточно строгие требования к условиям съемки и нуждается в надежном механизме нахождения ключевых точек для общего случая. Кроме того, требуется применение более совершенных методов классификации или построения модели изменений. В общем случае этот метод не является самым оптимальным, однако для некоторых специфических задач все же перспективен. К таким задачам можно отнести документный контроль, когда требуется сравнить изображение лица, полученного в текущий момент, с фотографией в документе. При этом других изображений этого человека не имеется, и, следовательно, механизмы классификации, основанные на анализе тренировочного набора, недоступны.



Идентификационные точки и расстояния: *а* - используемые при криминалистической экспертизе; *б* - наиболее часто применяемые при построении автоматизированных систем идентификации

**Сравнение шаблонов**.

Сравнение шаблонов (Template Matching) заключается в выделении областей лица на изображении рис. 5, и последующем сравнении этих областей для двух различных изображений. Каждая совпавшая область увеличивает меру сходства изображений. Это также один из исторически первых методов распознавания человека по изображению лица. Для сравнения областей используются простейшие алгоритмы вроде попиксельного сравнения.

Недостаток этого метода заключается в том, что он требует много ресурсов как для хранения участков, так и для их сравнения. Ввиду того, что используется простейший алгоритм сравнения, изображения должны быть сняты в строго установленных условиях: не допускается заметных изменений ракурса, освещения, эмоционального выражения и пр.



Области, входящие в шаблон лица

Скрытые Марковские модели.

Марковские модели являются мощным средством моделирования различных процессов и распознавания образов. По своей природе Марковские модели позволяют учитывать непосредственно пространственно-временные характеристики сигналов, и поэтому получили широкое применение в распознавании речи, а в последнее время - изображений (в частности, изображений лиц). Каждая модель , представляет собой набор состояний , между которыми возможны переходы. В каждый момент времени система находится в строго определенном состоянии. В наиболее распространенных Марковских моделях *первого* *порядка* первого порядка полагается, что следующее состояние зависит только от текущего состояния. При переходе в каждое состояние генерируется наблюдаемый символ, который соответствует физическому сигналу с выхода моделируемой системы. Набор символов для каждого состояния , количество символов . Выход, генерируемый моделью, может быть так же непрерывным. Существуют так же модели, в которых набор символов для всех состояний одинаков. Символ в состоянии в момент времени генерируется с вероятностью . Набор всех таких вероятностей составляет матрицу .

Матрица определяет вероятность перехода из одного состояния в другое: . Считается, что не зависит от времени. Если из каждого состояния можно достичь любого другого за один переход, то все , и модель называется эргодической. Модель имеет вероятность начальных состояний , где . Обычно в реальных процессах последовательность состояний является скрытой от наблюдения и остается неизвестной, а известен только выход системы, последовательность наблюдаемых символов , где каждое наблюдение - символ из , и - число наблюдений в последовательности. Поэтому такие модели называют *скрытыми* Марковскими моделями (Hidden Markov Models, ).

Модель с настроенными параметрами может быть использована для генерирования последовательности наблюдений. Для этого случайно, в соответствии с начальными вероятностями выбирается начальное состояние, затем на каждом шаге вероятность используется для генерации наблюдаемого символа, а вероятность - для выбора следующего состояния. Вероятность генерирования моделью последовательности состояний :

Где - последовательность состояний. Предполагается, что наблюдения статистически независимы.

В распознавании образов скрытые Марковские модели применяются так. Каждому классу соответствует своя модель . Распознаваемый образ (речевой сигнал, изображение и т. д.) представляется в виде последовательности наблюдений . Затем для каждой модели вычисляется вероятность того, что эта последовательность могла быть сгенерирована именно этой моделью. Модель , получившая наибольшую вероятность, считается наиболее подходящей, и образ относят к классу .

В связи с этим появляются несколько вопросов, называемых тремя основными задачами скрытых Марковских моделей.

1. Имея последовательность наблюдений

и настроенную модель , как оценить вероятность генерации этой моделью данной последовательности наблюдений? Эта задача называется задачей распознавания.

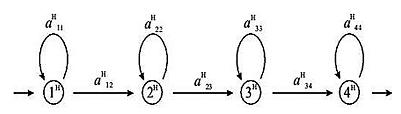
1. Имея последовательность наблюдений

и настроенную модель , как подобрать последовательность состояний , чтобы она была оптимальной (в соответствии с некоторым критерием, аналитически эта задача неразрешима)? Другими словами, это задача объяснения. Она нужна для последующей коррекции параметров модели.

1. Каким образом корректировать параметры модели

, для того чтобы максимизировать ? То есть как сделать так, чтобы модель больше соответствовала своему классу, одним из образов которого является данная последовательность наблюдений (или несколько различных последовательностей)? Это задача обучения.

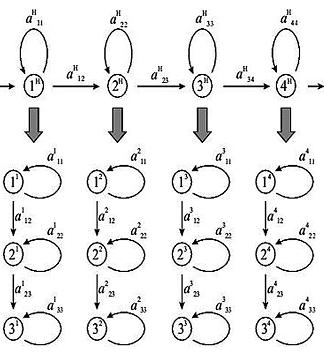
Первая задача имеет точное аналитическое решение, называемое процедурой прямого-обратного прохода. Последующие две задачи не имеют точного аналитического решения. Для решения второй задачи используется алгоритм Витерби, для третей - алгоритм Баума-Вельча. Оба этих метода являются разновидностями градиентного спуска и решаются оптимизационными методами.



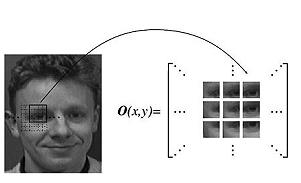
Линейная Марковская модель

Для того чтобы сократить вычисления, в распознавании речи используются линейные модели рис. 6. В таких моделях каждое состояние имеет только одно последующее, так же переход возможен обратно в то же состояние. Такие модели учитывают временные характеристики речевого сигнала: определенный порядок следования участков сигнала, их взаимное расположение, возможность локальных растяжений или сжатий. Это позволяет их применять и в распознавании изображений.

Суть двумерных Марковских моделей заключается в том, что, в отличие от одномерных линейных СММ, они позволяют моделировать искажения изображения и взаимное расположение участков не отдельно по горизонтали или вертикали, а в обоих направлениях одновременно. Для уменьшения вычислительной сложности применяются псевдодвумерные СММ (Pseudo-2D Hidden Markov Models, P2D-HMM). Такая модель состоит из нескольких линейных вертикальных моделей нижнего уровня и одной линейной горизонтальной модели верхнего уровня, на вход которой поступают выходы моделей нижнего уровня, рис. 7. Каждое



Псевдодвумерная скрытая Марковская модель



Извлечение участков-образцов наблюдения

состояние модели верхнего уровня включает в себя последовательность состояний соответствующей модели нижнего уровня. Модели нижнего уровня не связаны между собой. Изначально в модели верхнего уровня были вертикальными. В последующих работах модели верхнего уровня были сделаны горизонтальными (как это и изображено на рисунке), для того чтобы вертикальные модели нижнего уровня могли учесть тот факт, что глаза могут находиться на разной высоте. Таким образом, псевдодвумерная модель позволяет учесть локальные деформации и взаимное расположение участков изображений. Но в отличие от оптических потоков и других методов сопоставления деформациями, псевдодвумерная модель учитывает характер деформаций, а то, какими именно могут быть возможные деформации, псевдодвумерные СММ усваивают в процессе обучения. Другими словами, участок, соответствующий глазу, никогда не будет сопоставлен, например, участку на месте рта.

Пример функционирования СММ. Входом СММ являются квадратные участки изображений (рис. 8). Было обнаружено, что участки, извлекаемые с 75{\%} перекрытием друг с другом, дают наилучшую точность распознавания.

Для СММ важное значение имеет начальная инициализация модели. В качестве начальной инициализации всех моделей используются все изображения из тренировочного набора. Затем модель каждого класса настраивается на свои изображения.

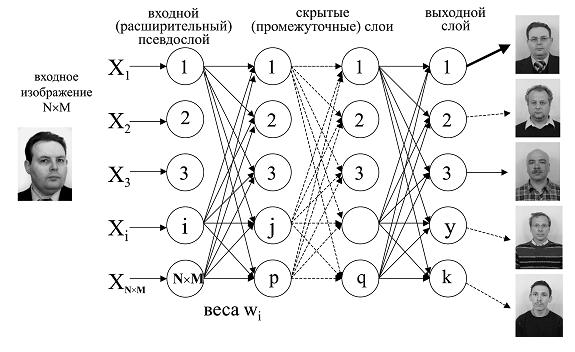
Недостатком СММ является то, что СММ не обладает различающей способностью, т. е. алгоритм обучения только максимизирует отклик каждой модели на свои классы, но не минимизирует отклик на другие классы и не выделяются ключевые признаки, отличающие один класс от другого. Таким образом, похожие классы могут оказаться слабо различимыми и при увеличении объема базы или использования в более широких условиях СММ может оказаться ненадежными.

Многослойные нейронные сети.

Архитектура многослойной нейронной сети (МНС) состоит из последовательно соединенных слоев, где нейрон каждого слоя своими входами связан со всеми нейронами предыдущего слоя, а выходами - следующего. НС с двумя решающими слоями может с любой точностью аппроксимировать любую многомерную функцию. НС с одним решающим слоем способна формировать линейные разделяющие поверхности, что сильно сужает круг задач, ими решаемых, в частности, такая сеть не сможет решить задачу типа "исключающее или". НС с нелинейной функцией активации и двумя решающими слоями позволяет формировать любые выпуклые области в пространстве решений, а с тремя решающими слоями - области любой сложности, в том числе и невыпуклой. При этом МНС не теряет своей обобщающей способности. Обучаются МНС при помощи алгоритма обратного распространения ошибки, являющегося методом градиентного спуска в пространстве весов с целью минимизации суммарной ошибки сети. При этом ошибки (точнее, величины коррекции весов) распространяются в обратном направлении от входов к выходам, сквозь веса, соединяющие нейроны. Простейшее применение однослойной НС (называемой автоассоциативной памятью) заключается в обучении сети восстанавливать подаваемые изображения. Подавая на вход тестовое изображение и вычисляя качество реконструированного изображения, можно оценить, насколько сеть распознала входное изображение. Положительные свойства этого метода заключаются в том, что сеть может восстанавливать искаженные и зашумленные изображения, но для более серьезных целей он не подходит. МНС также используется для непосредственной классификации изображений - на вход подается или само изображение в каком-либо виде, или набор ранее извлеченных ключевых характеристик изображения, на выходе нейрон с максимальной активностью указывает принадлежность к распознанному классу (рис. 9).

Нейрон с максимальной активностью (здесь первый) указывает принадлежность к распознанному классу. Если эта активность ниже некоторого порога, то считается, что поданный образ не относится ни к одному из известных классов. Процесс обучения устанавливает соответствие подаваемых на вход образов с принадлежностью к определенному классу. Это называется "обучением с учителем". В применении к распознаванию человека по изображению лица такой подход хорош для задач контроля доступа небольшой группы лиц. Он обеспечивает непосредственное сравнение сетью самих образов, но с увеличением числа классов время обучения и работы сети возрастает экспоненциально, и поэтому для таких задач, как поиск похожего человека в большой базе данных, требует извлечения компактного набора ключевых характеристик, на основе которых можно производить поиск.

В частности, МНС может использоваться для классификации изображений лиц на основе таких характеристик, как расстояния между некоторыми специфическими частями лица (нос, рот, глаза). Также существуют гибридные системы, например объединение с марковской моделью. В классической МНС межслойные нейронные соединения полносвязны, и изображение представлено в виде одномерного вектора, хотя оно двумерно. Архитектура сверточной НС направлена на преодоление этих недостатков. В ней используются локальные рецепторные поля (обеспечивают локальную двумерную связность нейронов), общие веса (обеспечивают детектирование некоторых черт в любом месте изображения) и иерархическая организация с пространственными подвыборками (spatial subsampling). Сверточная НС (СНС) обеспечивает частичную устойчивость к изменениям масштаба, смещениям, поворотам, искажениям. Архитектура СНС состоит из многих слоев, каждый из которых имеет несколько плоскостей, причем нейроны следующего слоя связаны только с небольшим числом нейронов предыдущего слоя из окрестности локальной области (как в зрительной коре человека). Веса в каждой точке одной плоскости одинаковы (сверточные слоя). За сверточным слоем следует слой, уменьшающий его размерность путем локального усреднения. Затем опять сверточный слой, и так далее. Таким образом, достигается иерархическая организация. Более поздние слои извлекают более общие характеристики, меньше зависящие от искажений изображения. Обучается СНС стандартным методом обратного распространения ошибки. Сравнение МНС и СНС показало существенные преимущества последней как по скорости, так и по надежности классификации. Полезным свойством СНС является и то, что характеристики, формируемые на выходах верхних слоев иерархии, могут быть применимы для классификации по методу ближайшего соседа (например, вычисляя евклидово расстояние), причем СНС может успешно извлекать такие характеристики и для образов, отсутствующих в обучающем наборе. Для СНС характерны высокая скорость обучения и работы. Тестирование СНС на базе данных ORL, содержащей изображения лиц с небольшими изменениями освещения, масштаба, пространственных поворотов, положения и различными эмоциями, показало приблизительно 98{\%} точность распознавания, причем для известных лиц предъявлялись варианты их изображений, отсутствующие в обучающем наборе. Такой результат делает эту архитектуру перспективной для дальнейших разработок в области распознавания изображений пространственных объектов. МНС применяются и для обнаружения объектов определенного типа. Помимо этого, любая обученная МНС в некоторой мере может определять принадлежность образов к "своим" классам, ее можно специально обучить надежному детектированию определенных классов. В этом случае выходными классами будут классы, принадлежащие и не принадлежащие к заданному типу образов.



Многослойная нейронная сеть для классификации изображений

Сети Габоровых вейвлетов (GWN).

Данный метод предназначен для слежения за лицом в реальном времени с помощью Gabor wavelet template (GWT), представляющего собой дискретную линейную комбинацию Габоровых вэйвлетов. Важно отметить, что точность представления лица регулируется числом Габоровых вэйвлетов в GWN, допуская представления множества лиц одним GWT. Такое представление допускает произвольные аффинные преобразования и быструю оценку аффинных параметров методом градиентного спуска. Тем самым, выполняя слежение за лицом, метод определяет и его ориентацию, однако с ограничением на повороты: допускаются, в основном, повороты в плоскости и незначительные пространственные вокруг вертикальной оси.

Нейронные сети Хопфилда.

НС Хопфилда (НСХ) является однослойной и полносвязной (связи нейронов на самих себя отсутствуют), ее выходы связаны со входами. В отличие от МНС, НСХ является релаксационной - т. е. будучи установленной в некое начальное положение функционирует до тех пор, пока не достигнет стабильного состояния, которое и будет являться ее выходным значением. НСХ применяются в качестве ассоциативной памяти и для решения оптимизационных задач. В первом случае НСХ обучается без учителя (например, по правилу Хебба), во втором случае веса между нейронами изначально кодируют решаемую задачу. НСХ бывают синхронными, когда одновременно пересчитываются все нейроны, и асинхронными, когда пересчитывается случайно выбранный нейрон. Для исследования динамики функционирования НСХ используются методы Ляпунова. Асинхронная НСХ всегда сходится к устойчивым точкам, а аттракторами синхронной НСХ являются устойчивые стационарные точки и предельные циклы длины два. Таким образом, НСХ из начального состояния сходится к ближайшему локальному минимуму энергии сети, состояние нейронов в котором и будет восстановленным образом для задач распознавания, и решением - для оптимизационных задач. Для поиска глобального минимума применительно к оптимизационным задачам используют стохастические модификации НСХ.

Применение НСХ в качестве ассоциативной памяти позволяет точно восстанавливать образы, которым сеть обучена, при подаче на вход искаженного образа. При этом сеть "вспомнит" наиболее близкий (в смысле локального минимума энергии) образ, и распознает его. Такое функционирование также можно представить как последовательное применение автоассоциативной памяти. В отличие от автоассоциативной памяти, НСХ идеально точно восстановит образ.

Хотя разработка систем идентификации по изображениям человеческих лиц ведется уже несколько десятилетий, задача создания эффективного алгоритма идентификации по изображениям человеческих лиц еще далека от завершения.

Для большинства современных систем автоматического распознавания лиц основной задачей является задача сравнения данного изображения лица с набором изображений лиц из базы данных. Характеристики систем автоматического распознавания лиц в этом случае оцениваются путем определения вероятностей ошибочного отказа в распознавании (для изображения лица, присутствующего в базе, принимается решение как о неопознанном лице) и ошибочного распознавания. В дополнение к вероятностям ошибок для оценки системы автоматического распознавания лиц часто используется оценка устойчивости к возмущению изображений, вызываемая комбинацией со сложными фонами, изменчивостью освещения, изменению прически, и т. д.

Выбор алгоритма или группы алгоритмов идентификации лиц для создания практических систем автоматического распознавания лиц должен основываться на системе оценок рабочих характеристик конкретной системы, соответствующей ограничениям, связанным с условиями эксплуатации данной конкретной системы, и определяется непосредственно на этапе проектирования.

ДЕТЕКТИРОВАНИЕ ЧЕЛОВЕЧЕСКОГО ЛИЦА НА ИЗОБРАЖЕНИИ ДЛЯ ВХОДА СВЕРТОЧНОЙ СЕТИ

Для детектирования лица используется метод, называемый метод Виолы-Джонса, также называемый каскадом Хаара.

Метод был разработан и представлен в 2001 г. Полом Виолой и Майклом Джонсом. Он до сих пор является эффективным методом для поиска объектов на изображениях и видеопоследовательностях в режиме реального времени [1, 2]. Следует отметить, что этот детектор обладает крайне низкой вероятностью ложного обнаружения лица. Метод хорошо работает и обнаруживает черты лица даже при наблюдении объекта под небольшим углом, примерно до 30°. При угле наклона больше 30° вероятность обнаружения лица резко падает. Указанная особенность метода не позволяет в стандартной реализации детектировать лицо человека, повернутое под произвольным углом, что в значительной мере затрудняет или делает невозможным использование алгоритма в современных производственных системах с учетом их растущих потребностей.

Интегральное представление изображений

Для того, чтобы рассчитать яркость прямоугольного участка изображения, используют интегральное представление [3]. Такое представление используется часто и в других методах, например, в вейвлет преобразованиях, Speeded up robust feature (SURF), фильтрах Хаара и многих разработанных алгоритмах.

Интегральное представление позволяет быстро рассчитывать суммарную яркость произвольного прямоугольника на данном изображении, причем время расчета не зависит от площади прямоугольника. Интегральное представление изображения представляет собой матрицу, совпадающую по размерам с исходным изображением. В каждом ее элементе хранится сумма интенсивностей всех пикселей, находящихся левее и выше данного элемента. Элементы матрицы рассчитываются по следующей формуле:

где – значение точки интегрального изображения; – значение интенсивности исходного изображения. На основе применения интегрального представления изображения вычисление признаков одинакового вида, но с разными геометрическими параметрами, происходит за одинаковое время.

Каждый элемент матрицы представляет собой сумму пикселей в прямоугольнике от до , т. е. значение каждого элемента равно сумме значений всех пикселей левее и выше данного пикселя . Расчет матрицы занимает линейное время, пропорциональное числу пикселей в изображении и его можно производить по следующей формуле:

Интегральное представление имеет интересную особенность. По интегральной матрице можно очень быстро вычислить сумму пикселей произвольного прямоугольника.

Хаар-подобные характеристики

С точки зрения необходимости использования достаточно простых алгоритмов получения признаков, перспективным является использование Хаар-подобных характеристик, представляющих собой результат сравнения яркостей в двух прямоугольных областях изображения.

Предположим, что задано множество объектов A и множество допустимых ответов B. Пусть g:A→B называется решающей функцией. Решающая функция g должна допускать эффективную компьютерную реализацию, по этой причине её также называют алгоритмом. Признак (feature) f объекта a – отображение f:A→Df где Df – множество допустимых значений признака. В частности, любой алгоритм g:A→B также можно рассматривать как признак. Если задан набор признаков f1,…, fn, то вектор x=(f1(a),…,fn(a)) называется признаковым описанием объекта a∈A. Признаковые описания допустимо отождествлять с самими объектами. При этом множество A=Df1×…×Dfn называют признаковым пространством [4].

В стандартном методе Виолы–Джонса используются прямоугольные признаки, рис. 1. Эти признаки называются примитивами Хаара.

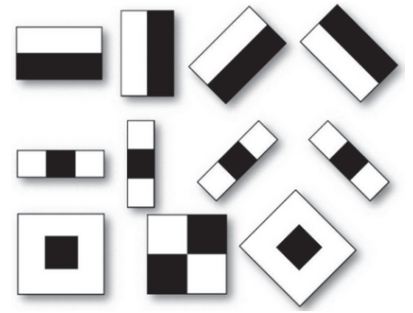


Рис. 1.

Примитивы признаков Хаара

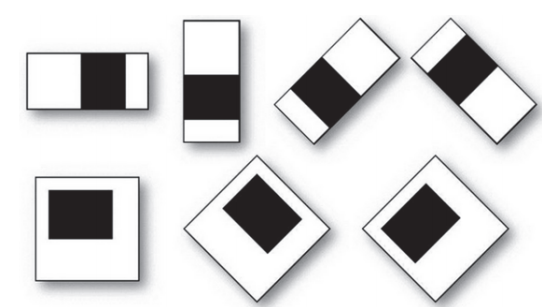


Рис. 2. Дополнительные признаки Хаара

В расширенном методе Виолы–Джонса, представленном в библиотеке OpenCV, используются дополнительные признаки (рис. 2).

Вычисляемым значением такого признака будет:

где – сумма значений яркостей точек, закрываемых светлой частью признака, а – сумма значений яркостей точек, закрываемых темной частью признака. Для их вычисления используется понятие интегрального изображения. Хаар-подобные признаки описывают значение перепада яркости по оси X и Y изображения соответственно.

Метод построения классификатора на основе алгоритма бустинга

Бустинг – комплекс методов, способствующих повышению точности аналитических моделей. Бустинг (boosting) означает дословно «усиление» «слабых» моделей [5] – это процедура последовательного построения композиции алгоритмов машинного обучения, когда каждый следующий алгоритм стремится компенсировать недостатки композиции всех предыдущих алгоритмов. Идея бустинга была предложена Робертом Шапиро (Schapire) в конце 90 х гг. прошлого века [6], когда надо было найти решение вопроса о том, каким образом имея множество плохих (незначительно отличающихся от случайных) алгоритмов обучения, получить один хороший.

В результате работы алгоритма бустинга на каждой итерации формируется простой классификатор вида:

где – показывает направление знака неравенства; – значение порога; – вычисленное значение признака; z – окно изображения размером 24×24 пикселей. Полученный классификатор имеет минимальную ошибку по отношению к текущим значениям весов, задействованным в процедуре обучения для определения ошибки.

Метод комбинирования классификаторов в каскадную структуру

Каскадная структура повышает скорость обнаружения, фокусируя свою работу на наиболее информативных областях изображения.

Структура каскадного детектора приведена на рис. 3. Каскад состоит из слоев, которые представляют собой классификаторы, обученные с помощью процедуры бустинга.

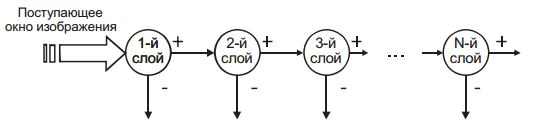


Рис. 3. Структура каскадного детектора

НАХОЖДЕНИЕ ТОЧЕК ЛИЦА ПО ДЕТЕКТИРОВАННОМУ СЕКТОРУ

СТРУКТУРА СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ

Сверточные нейронные сети содержат три типа слоев:

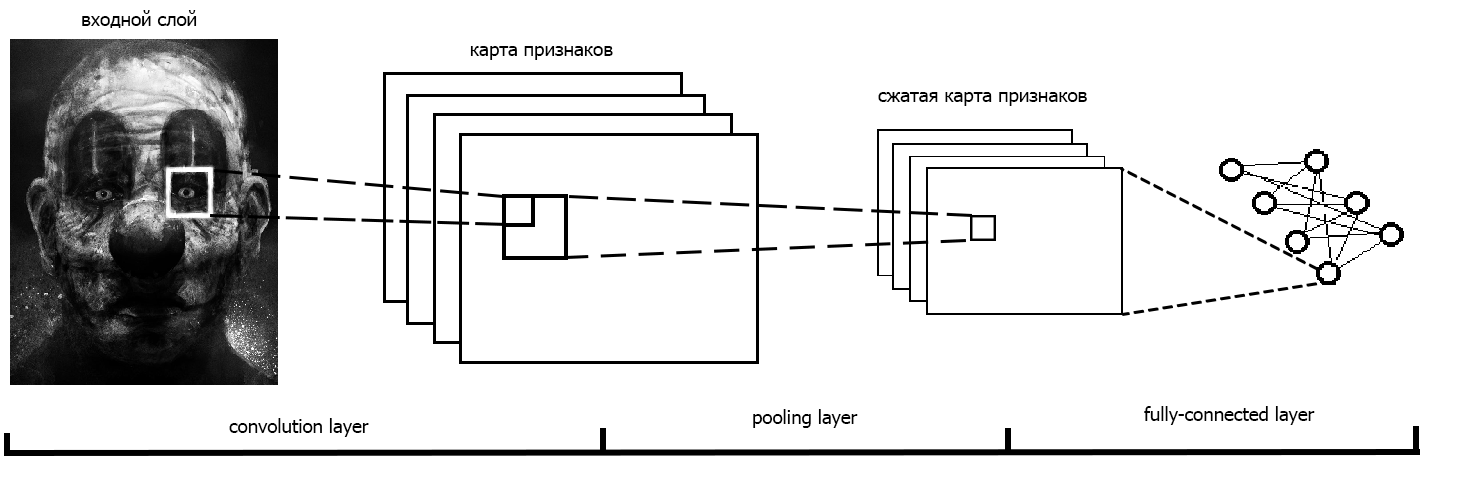
1. **Convolutional(сверточные).** В операции свертки используется лишь ограниченная матрица весов небольшого размера, которую «двигают» по всему обрабатываемому слою (в самом начале по входным данным), формируя после каждого сдвига сигнал активации для нейрона следующего слоя с аналогичной позицией. То есть для различных нейронов выходного слоя используются общие веса — матрица весов, которую также называют набором весов или ядром свертки. Она построена таким образом, что графически кодирует какой-либо один признак, например, наличие наклонной линии под определенным углом. Тогда следующий слой, получившийся в результате операции свертки такой матрицей весов, показывает наличие данной наклонной линии в обрабатываемом слое и ее координаты, формируя так называемую карту признаков (feature map). В сверточной нейронной сети количество ядер, как правило много: они кодируют признаки исходных данных.

Своё название свёрточная сеть получила по названию операции – свёртка, она часто используется для обработки изображений и может быть описана следующей формулой.

Здесь - исходная матрица изображения, – ядро, – координаты центра ядра,

При этом в зависимости от метода обработки краёв исходной матрицы результат может быть меньше исходного изображения (valid), такого же размера (same) или большего размера (full).

1. **Pooling, subsampling(операция субдискретизации).** После каждого сверточного слоя, может быть pooling слой. Pooling слой берет небольшие прямоугольные блоки из сверточного слоя и вычисляет один выход из этого блока. Есть несколько способов сделать эту операцию, например, взяв среднее, максимальное или какую-либо линейную комбинацию нейронов в блоке.
2. **Fully-Connected(персептрон, MLP, полносвязная сеть).** После нескольких convolutional и max-pooling слоев следуют полносвязные слои. Fully connected слой принимает все нейроны из предыдущего слоя (будь то fully connected, pooling или convolutional) и соединяет его с каждым нейроном, который он имеет. Fully connected слои не могут быть расположены где угодно, то есть не может быть сверточных слоев после того, как идет fully connected слой.



Первые два типа слоёв (convolutional, pooling), чередуясь между собой, формируют входной вектор признаков для fully-connected. Fully-connected слой по вектору признаков выдает обобщенный результат. Сеть можно обучать с помощью градиентных методов.

Применение нейронных сетей не ограничивается двумерным случаем. Возможно точно таким же образом построить одно или трехмерные сверточные нейронные сети; convolutional фильтры будут просто иметь размеры определенные надлежащим образом, и pooling слои изменят измерение. Например, возможно использовать одномерные сверточные сети на аудио данных.

ПРИНЦИП РАБОТЫ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ

Convolution слой

Свёрточный слой реализует идею локальных рецептивных полей, то есть каждый выходной нейрон соединен только с определённой областью входной матрицы и таким образом моделирует некоторые особенности человеческого зрения.

Этот слой описывается следующий формулой:

Здесь - выход слоя , - функция активации, - коэффициент сдвига, символом ∗ обозначена операция свёртки входа с ядром .

При этом за счёт краевых эффектов размер исходных матриц уменьшается.

Здесь *xj* - карта признаков *j* (выход слоя *l*), *f()* - функция активации, *b**j* - коэффициент сдвига для карты признаков *j*, *kj* - ядро свёртки номер *j*, *xil*−1 - карты признаков предыдущего слоя.

Pooling слой

Слои этого типа выполняют уменьшение размера входной карты признаков. Это можно делать разными способами, в данном случае мы рассмотрим метод выбора максимального элемента (max-pooling) - вся карта признаков разделяется на ячейки 2х2 элемента, из которых выбираются максимальные по значению. Формально слой может быть описан следующим образом.

Здесь *xl* - выход слоя *l*, *f()* - функция активации, *a,b* - коэффициенты, *subsample()* - операция выборки локальных максимальных значений.

Использование этого слоя позволяет улучшить распознавание образцов с изменённым масштабом (уменьшенных или увеличенных).

Fully-connected

Последний из типов слоёв это слой многослойного перцептрона (MLP), его можно описать следующим соотношением.

Здесь *xl* - выход слоя *l*, *f()* - функция активации, *b* - коэффициент сдвига, *w* - матрица весовых коэффициентов.

ПРИНЦИП ОБУЧЕНИЯ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ

Для обучения свёрточной сети применяются градиентные методы

Вычисление ошибки

**Для выходного (MLP)** слоя ошибка рассчитывается следующим образом.

Здесь *T* - ожидаемый (учебный) выход , *Y* - реальный выход, *f*′*(u)* - производная функции активации по её аргументу

**Для скрытых слоёв MLP** ошибка имеет следующий вид.

Здесь *δ l*- ошибка слоя *l*, *f*′*(ul)* - производная функции активации, *ul* - состояние (не активированное) нейронов слоя *l*, *Wl* - матрица весовых коэффициентов слоя *l*.

**Ошибка на выходе свёрточного слоя** формируется путём простого увеличения размера матриц ошибки следующего за ним субдискретизирующего слоя.

Здесь *δl* - ошибка слоя *l*, *f*′*(ul)* - производная функции активации, *ul* - состояние (не активированное) нейронов слоя *l*,*upsample()* - операция увеличения размера матриц.

**Ошибка на выходе субдискретизирующего** слоя рассчитывается путём выполнения ”обратной свёртки” карт признаков следующего за ним свёрточного слоя, т.е. над каждой картой признаков выполняется свёртка с соответствующим ”перевернутым” ядром, при этом за счёт краевых эффектов размер исходных матриц увеличивается. Далее над получившимися картами вычисляются несколько частичных сумм по числу ядер свертки, в соответствии с матрицей смежности субдискретизирующего и свёрточного слоёв.

Здесь *δl* - ошибка слоя *l*, *f*′*(ul)* - производная функции активации, *ul* - состояние (не активированное) нейронов слоя *l*, *k* - ядра свёртки.

Вычисление градиента

В этом разделе описана процедура вычисления градиента функции потери сети, т.е. направления максимального роста функции потери. Обучение сводиться к её минимизации в пространстве параметров (весов) сети.

* Градиент для ядра свёртки можно посчитать как свёртку матрицы входа свёрточного слоя с "перевёрнутой" матрицей ошибки для выбранного ядра.

Здесь *δl* - ошибка слоя *l*, *xl* - вход слоя *l*, *k* - ядра свёртки.

Градиент для сдвига для свёрточного слоя вычисляется как сумма значений соответствующей матрицы ошибки.

Здесь *δl* - ошибка слоя *l*

* Градиент для коэффициентов субдискретизирующего слоя вычисляется следующим образом.

Здесь *xl* - выход слоя *l*, *δl* - ошибка слоя *l*, *subsample()* - операция выборки локальных максимальных значений.

Градиент для коэффициента сдвига для субдискретизирующего слоя вычисляется как сумма значений соответствующей матрицы ошибки.

Здесь *δl* - ошибка слоя *l*

* Градиент для весов MLP выглядит следующим образом.

Здесь *δl* - ошибка слоя *l*, *xl* - вход слоя *l*, *Wl* - матрица весовых коэффициентов слоя *l*.

АРХИТЕКТУРА СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ В ДАННОЙ РАБОТЕ

input (None, 3, 100, 100) produces 30000 outputs

conv1 (None, 32, 98, 98) produces 307328 outputs

pool1 (None, 32, 49, 49) produces 76832 outputs

dropout1 (None, 32, 49, 49) produces 76832 outputs

conv2 (None, 64, 48, 48) produces 147456 outputs

pool2 (None, 64, 24, 24) produces 36864 outputs

dropout2 (None, 64, 24, 24) produces 36864 outputs

conv3 (None, 128, 23, 23) produces 67712 outputs

pool3 (None, 128, 12, 12) produces 18432 outputs

dropout3 (None, 128, 12, 12) produces 18432 outputs

hidden4 (None, 500) produces 500 outputs

dropout4 (None, 500) produces 500 outputs

hidden5 (None, 500) produces 500 outputs

output (None, 16) produces 16 outputs

МЕТОДЫ ПРИМЕНЯЕМЫЕ ПРИ ОБУЧЕНИИ

Обучение на GPU

Важную роль в процессе обучения нейронных сетей занимает время ее обучения, так как современные сети как правило очень сложны и без какой-либо оптимизации они могут обучаться годами. Один из способов ускорить обучение – это распараллеливание процессов.

В сверточных сетях, есть независимые операции(например вычисление по различных фильтрам соответствующие им future map)

В глубоком обучение активно используется обучение на видеокартах, так как они обладают множеством ядер.

Специалисты по обработке и анализу данных как в промышленности, так и в научных кругах используют GPU в сфере машинного обучения, чтобы добиться значительных усовершенствований в широком спектре приложений, включая приложения для классификации изображений, анализа видеоданных, распознавания речи и обработки текстов на естественном языке. Глубокое обучение, то есть использование сложных, многоуровневых нейронных сетей для создания систем, которые могут выявлять признаки из большого объема немаркированных данных, - именно та область, в которой ведутся активные исследования и инвестиционная деятельность.

Хотя машинное обучение существует уже десятки лет, две относительно новые тенденции привели к его широкомасштабному использованию: доступность большого объема данных, а также производительность и эффективность параллельной обработки данных, которая возможна благодаря [вычислениям на GPU](http://www.nvidia.ru/object/gpu-computing-ru.html). GPU используются для обучения этих глубоких нейронных сетей с помощью намного более крупных обучающих последовательностей в более сжатые сроки, с использованием меньшей инфраструктуры ЦОД. GPU также используются, чтобы воспроизводить эти учебные модели машинного обучения для выполнения задач классификации и прогнозирования на облаке. При этом графические процессоры позволяют работать с данными большего объема и с более высокой производительностью, потребляя меньше энергии и на базе меньшей инфраструктуры.

К числу тех, кто впервые применил графические ускорители для решения задач машинного обучения, относятся многие крупные веб-компании и социальные сетевые серверы, наряду с научно-исследовательскими институтами высокого ранга в области обработки и анализа данных и машинного обучения. Благодаря тысячам вычислительных ядер и увеличению производительности приложений в 10-100 раз по сравнению с CPU, GPU стали процессорами, которые выбирают специалисты по обработке данных для работы с данными большого объема.

Увеличение тестовой выборки

Зачастую в глубоком обучение наибольшую выгоду приносит простое увеличение тестовой выборки.

Предотвращение переобучения

Переобучение, переподгонка (overtraining, overfitting) — нежелательное явление, возникающее при решении задач [обучения по прецедентам](http://www.machinelearning.ru/wiki/index.php?title=%D0%9E%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5_%D0%BF%D0%BE_%D0%BF%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82%D0%B0%D0%BC), когда вероятность ошибки обученного алгоритма на объектах [тестовой выборки](http://www.machinelearning.ru/wiki/index.php?title=%D0%A2%D0%B5%D1%81%D1%82%D0%BE%D0%B2%D0%B0%D1%8F_%D0%B2%D1%8B%D0%B1%D0%BE%D1%80%D0%BA%D0%B0) оказывается существенно выше, чем средняя ошибка на [обучающей выборке](http://www.machinelearning.ru/wiki/index.php?title=%D0%9E%D0%B1%D1%83%D1%87%D0%B0%D1%8E%D1%89%D0%B0%D1%8F_%D0%B2%D1%8B%D0%B1%D0%BE%D1%80%D0%BA%D0%B0). Переобучение возникает при использовании избыточно сложных моделей.

Для того, чтобы избежать чрезмерной подгонки, необходимо использовать дополнительные методы, например:

1. перекрёстная проверка

**Перекрёстная проверка** (англ *Cross-validation*) — метод оценки аналитической модели и её поведения на независимых данных. При оценке модели имеющиеся в наличии данные разбиваются на k частей. Затем на k−1 частях данных производится обучение модели, а оставшаяся часть данных используется для тестирования. Процедура повторяется k раз; в итоге каждая из k частей данных используется для тестирования. В результате получается оценка эффективности выбранной модели с наиболее равномерным использованием имеющихся данных.

1. регуляризация

**Регуляризация** в статистике, машинном обучении, теории обратных задач — метод добавления некоторой дополнительной информации к условию с целью решить некорректно поставленную задачу или предотвратить переобучение. Эта информация часто имеет вид штрафа за сложность модели. Например, это могут быть ограничения гладкости результирующей функции или ограничения по норме векторного пространства.

1. ранняя остановка
2. вербализация нейронных сетей

**Вербализация** — минимизированное описание работы синтезированной и уже обученной нейронной сети в виде нескольких взаимозависимых алгебраических или логических функций.

1. априорная вероятность,
2. байесовское сравнение моделей

которые могут указать, когда дальнейшее обучение больше не ведёт к улучшению оценок параметров. В основе этих методов лежит явное ограничение на сложность моделей, или проверка способности модели к обобщению путём оценки её эффективности на множестве данных, не использовавшихся для обучения и считающихся приближением к реальным данным, к которым модель будет применяться.

Dropout

Это техника уменьшения переобучения в нейронных сетях предотвращающая сложную адаптацию к тренировочных данным. Это очень эффективный путь для усреднения результатов нейронной сети. Термин dropout означает исключение некоторого количества случайных нейронов из нейронной сети на текущей итерации.

**Стохастический и пакетный режим обучения**

Локальные минимумы

Обратное распространение использует разновидность градиентного спуска, то есть осуществляет спуск вниз по поверхности ошибки, непрерывно подстраивая веса в направлении к минимуму. Поверхность ошибки сложной сети сильно изрезана и состоит из холмов, долин, складок и оврагов в пространстве высокой размерности. Сеть может попасть в локальный минимум (неглубокую долину), когда рядом имеется гораздо более глубокий минимум. В точке локального минимума все направления ведут вверх, и сеть неспособна из него выбраться. Основную трудность при обучении нейронных сетей составляют как раз методы выхода из локальных минимумов: каждый раз выходя из локального минимума снова ищется следующий локальный минимум тем же методом обратного распространения ошибки до тех пор, пока найти из него выход уже не удаётся.

Перемешивание обучающий примеров

Так как данные в выборке могут быть изначально в некоторой определенной последовательности(например в выборке данных мужчин и женщин, данные могут быть отсортированы по полу), то получая данные, по порядку их следования, модель лучше обучается по последним данным.

**Нормализация входных данных**

Паралич сети

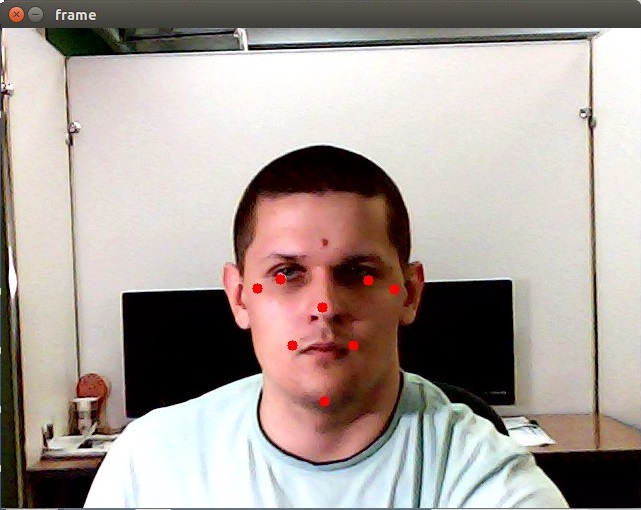
В процессе обучения сети значения весов могут в результате коррекции стать очень большими величинами. Это может привести к тому, что все или большинство нейронов будут функционировать при очень больших значениях OUT, в области, где производная сжимающей функции очень мала. Так как посылаемая обратно в процессе обучения ошибка пропорциональна этой производной, то процесс обучения может практически замереть. В теоретическом отношении эта проблема плохо изучена. Обычно этого избегают уменьшением размера шага η, но это увеличивает время обучения. Различные эвристики использовались для предохранения от паралича или для восстановления после него, но пока что они могут рассматриваться лишь как экспериментальные.

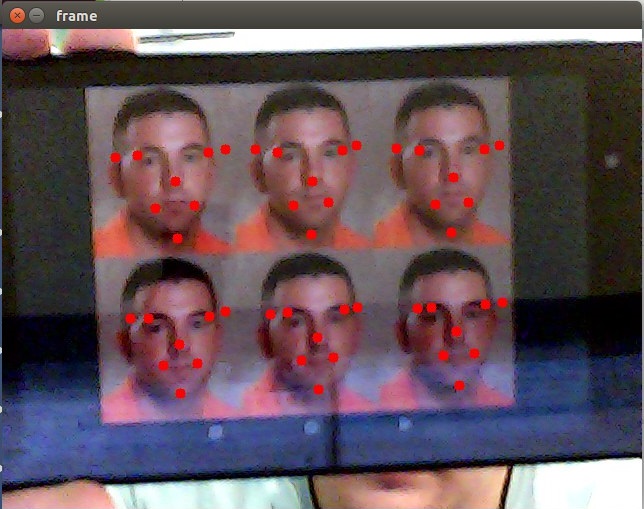
**Локальные минимумы**

**Функция активации**

**Функция потерь**

РЕЗУЛЬТАТЫ





ЗАКЛЮЧЕНИЕ

ЛИТЕРАТУРА

1. Hubel, D. H.; Wiesel, T. N. (1968-03-01)."Receptive fields and functional architecture of monkey striate cortex". *The Journal of Physiology* **195** (1): 215–243.ISSN 0022-3751.PMC 1557912. PMID 4966457.
2. LeCun, Yann; Léon Bottou; Yoshua Bengio; Patrick Haffner (1998)."Gradient-based learning applied to document recognition"
3. <http://www.deeplearningbook.org/>
4. Viola P., Jones M.J. Rapid object detection using a boosted cascade of simple features // IEEE Conf. on Computer Vision and Pattern Recognition. – Kauai, Hawaii, USA, 2001. – V. 1. – P. 511–518. 2.
5. Viola P., Jones M.J. Robust realtime face detection // International Journal of Computer Vision. – 2004. – V. 57. – № 2. – P. 137–154.
6. <http://cogprints.org/5869/1/cnn_tutorial.pdf>
7. <http://cs231n.github.io/>
8. <http://neuralnetworksanddeeplearning.com/chap6.html>
9. <http://ufldl.stanford.edu/tutorial/supervised/ConvolutionalNeuralNetwork/>
10. <http://andrew.gibiansky.com/blog/machine-learning/convolutional-neural-networks/>
11. <http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/>
12. <http://mechanoid.kiev.ua/ml-lenet.html>
13. Борисов Е. О методах обучения многослойных нейронных сетей прямого распространения.   
    – <http://mechanoid.kiev.ua/neural-net-backprop.html>
14. <http://mechanoid.kiev.ua/ml-lenet.html>

ПРИЛОЖЕНИЕ

Текст программы на языке python