**Исследование методов быстрого поиска изображений в задаче распознавания лиц**

**1. Актуальность**

В последнее десятилетие визуальный поиск стал широко распространенной функцией многих поисковых систем. Большой объем сохраненных изображений в сети измеряется петабайтами, число которых с каждым днем только увеличивается. В связи с этим эффективный поиск ближайших соседей стал серьезной исследовательской проблемой. Потребность быстрого поиска похожих изображений занимает большую нишу в современных приложениях компьютерного зрения, в том числе в задаче распознавания лиц. Социальные сети, правоохранительные органы имеют огромные коллекции изображений лиц, среди которых надо уметь быстро извлекать нужную информацию. Для решения данной проблемы требуются эффективные и масштабируемые алгоритмы поиска с низкими временными затратами. Ожидается, что ответ на запросы к базам данных из миллиардов элементов будет занимать несколько миллисекунд.

**2. Постановка**

Главной задачей данной работы является исследование современных индексных структур быстрого поиска и проверка их эффективности в задаче распознавания лиц. Будем считать, что выделением лица на изображении и построением его признаков занимаются алгоритмы общедоступных сверточных нейронных сетей.

Проверка заключается в измерении среднего времени поиска похожих лиц в большой коллекции изображений. Время, затраченное на поиск k (k = 1, 5, 10, 30, 50, 100) ближайших соседей будет основным критерием скорости алгоритма. Помимо скорости надо учитывать точность поиска. Во всех приведенных статьях точность измерялась как процент истинных ближайших соседей среди k найденных, где истинные ближайшие соседи определялись точным евклидовым расстоянием. В связи со спецификой нашей задачи, наиболее правильным вариантом измерения точности будет получение процента лиц, совпавших с лицом запрос, среди k найденных ближайших соседей. Также в качестве альтернативного показателя качества будем измерять частоту вхождения каждого лица в k ближайших соседей. В случае совпадения лица запроса и лица с наибольшей частотой будем говорить об успешном распознавании.

Для применимости данного подхода к реальным задачам требуемая скорость поиска не должна превышать нескольких миллисекунд. А точность должна быть в пределе допустимой для выбранного алгоритма, то есть не сильно отличаться от приводимой в статьях. В качестве вывода следует оценить эти показатели и сформулировать возможные пути улучшения.

**3. Существующие методы**

Начнем с того, что во всех упомянутых дальше алгоритмах решается одна и та же задача. А именно, вычисление евклидовых расстояний между векторами высокой размерности. Для сокращения времени поиска было предложено несколько методов многомерного индексирования, таких как популярное KD-дерево. Однако для больших размерностей оказывается, что такие подходы не намного эффективнее, чем исчерпывающий расчет расстояний. Также одним из самых популярных алгоритмов приближенного поиска является локально-чувствительное хеширование (LSH). Однако этот подход не учитывает требования к памяти структуры индексации и использование памяти может быть даже выше, чем у исходных векторов. В связи с этим более подробно обсудим алгоритмы индексирования основанные на квантовании.

Общая идея в том, что инвертированый индекс содержит связные списки векторов, где каждый список является отображением некоторого вектора-центроида. На первом этапе запроса определяется центроид, затем по определенному этим вектором списку выполняется исчерпывающий поиск.

Поиск по данной структуре производится в два этапа:

1) поиск ближайшего к запросу центроида по короткому списку центроидов;

2) поиск по списку кандидатов соответствующих этому центроиду.

Стандартным методом обучения центроидов VQ является алгоритм кластеризации k-средних.

Повысить эффективность этапа поиска и обучения можно с помощью инвертированного иерархического индекса (HKM), который помимо первичного разбиения на ячейки Вороного разбивает пространство каждой ячейки повторно.

И еще одна идея использования индексов для быстрого поиска предлагает разбивать пространство векторов на несколько подпространств меньшей размерности и обучить каждое подпространство малой размерности отдельно. Это позволяет разбивать датасет на огромное количество ячеек и с помощью декартова произведения центроидов быстро получать доступ к ним. Данная структура называется инвертированным мульти-индексом (IMI)

Недостатки этих алгоритмов в том, что существуют значительные зависимости между различными подпространствами дескрипторов, построенных с помощью сверточных нейронных сетей. Для решения проблемы адаптации алгоритмов к коррелированным данным можно предварительно производить ортогональные преобразования над данными (c) или локальные оптимизации PQ (d).

**4. Мое решение**

В качестве данных для исследования я использовал набор данных для распознавания лиц VGGFace2. Потому что это один самых больших датасетов лиц и изображения в нем загружаются из поиска картинок Google и имеют различия в эмоции, позе, возрасте, освещенности и этнической принадлежности. Для построение дескрипторов лиц использовалась общедоступная сверточная нейронная сеть из библиотеки Python facerecognition. Она дала лучшие показатели точности распознавания для поиска по евклидову расстоянию. С ее помощью было построено более 150000 векторов признаков лиц, на которых проводились эксперименты. Все эксперименты проводились 1000 раз и затем результат усреднялся.

Для проверки индексных алгоритмов поиска в задаче распознавания лиц были написаны следующие структуры: Простейший поиск по точному расстоянию, для сравнения с остальными алгоритмами, Простая индексная структура, Иерархическая индексная структура и Мультииндексная структура.

**5. Эксперименты**

Начнем с того, что рассмотрим график точности поиска по точному евкливову расстоянию. На графике можно видеть 2 ломанные обозначенные как Search и Recognition. Ломанная Search обозначает процент правильно найденных лиц среди k ближайших соседей. Ломанная Recognition обозначает процент правильно распознанных людей, где в качестве метрики использовалась наибольшая частота встречаемости лица среди соседей k ближайших соседей.

Можно заметить и в чем мы убедимся позже, что вторая кривая находится в некоторой окрестности постоянного значения. Это означает, что с расширением объема поиска процент лиц соответствующих запросу держится на одном уровне. Search кривая явно убывает и связано это с тем, что при увеличении количества ближайших соседей вероятность попадания посторонних лиц в их число также увеличивается.

Индексные структуры дают похожую картину за исключением того, что точность поиска несколько падает. Особенно сильно заметно падение на Мультииндексной структуре.

Относительное ухудшение точности можно увидеть на следующем графике. Здесь зеленым цветом отмечена кривая поиска по точному евклидову расстоянию.

Если говорить про время поиска, то из левого графика можно сделать вывод, что любая индексная структура ускоряет поиск в сотни раз. На правом графике изображены ломанные соответствующие времени поиска индексных структур для различных значений k ближайших соседей. Основная тенденция ломанных идет на возрастание, что связанно с тем, что при увеличении количества ближайших соседей также увеличивается список кандидатов.