

Санкт-Петербургский государственный университет

Математическое обеспечение и администрирование информационных
систем

Системное Программирование

Алиев Мирза Али оглы

Решение задачи кластеризации отпечатков пальцев методами глубокого обучения

Бакалаврская работа

Научный руководитель:
ст. преп. Сартасов С. Ю.

Рецензент:
Невоструев К. Н.

Санкт-Петербург
2016

SAINT-PETERSBURG STATE UNIVERSITY

Software and Administration of Information Systems
Software Engineering

Mirza Aliev

An approach for solving the problem of
clustering of fingerprints using deep learning
methods

Graduation Thesis

Scientific supervisor:
senior assistant professor Stanislav Sartasov

Reviewer:
Konstantin Nevostruev

Saint-Petersburg
2016

Оглавление

Введение	4
1. Постановка задачи	6
2. Исходные данные	7
3. Обзор существующих решений	8
3.1. Метрика сравнения алгоритмов	9
4. Методология	10
4.1. Вектор признаков	10
4.1.1. Поле направлений	11
4.1.2. Метод главных компонент	12
4.2. Кластеризация	12
4.2.1. Агломеративная иерархическая кластеризация . .	13
4.2.2. Метод k-средних	14
4.3. Глубокое обучение	14
5. Эксперименты	16
6. Сравнение с аналогами	18
Заключение	19
Список литературы	20

Введение

На сегодняшний день отпечаток пальца является основным и самым точным биометрическим идентификатором личности человека. Основано это на том, что каждый человек имеет уникальный рисунок папиллярных линий на пальцах.

Каждый день огромное количество отпечатков пальцев собирается и обрабатывается различными организациями, как коммерческими, так и государственными. В качестве примера можно привести Индию, в которой, в рамках программы Aadhaar по сбору биометрических данных жителей страны и выдаче уникальных идентификационных номеров, было собрано уже около миллиарда отпечатков пальцев. Также многие компании используют отпечатки пальцев для идентификации своих сотрудников.

Для выяснения, принадлежит ли отпечаток пальца базе, в худшем случае придется сравнивать исходный отпечаток со всеми отпечатками из базы. Вследствие больших размеров баз, данная задача может занять огромное количество времени, однако от систем идентификации человека требуется быстрое принятие точных решений. Под точностью понимается то, что система должна идентифицировать человека, чей отпечаток есть в базе, и наоборот, отклонить, если человека в базе нет.

Общепринятым способом увеличения скорости поиска отпечатка является разделение базы на классы. Вместо того, чтобы искать отпечаток по всей базе, отпечаток ищется в части базы, в соответствии с тем, к какому классу отпечаток относится.

Стоит отметить, что разделение базы на классы может осуществляться по-разному. Исторически сложилось, что самой популярной классификацией отпечатков является классификация Гальтона-Генри. Она основана на различных рисунках папиллярных линий, всего выделяется пять видов рисунков. Однако у данного подхода есть недостаток, заключается он в том, что отпечатки пальцев людей распределяются неравномерно на эти пять классов, на один из классов приходится примерно 30% отпечатков на выборке из 222 миллионов пальцев.

Из всего этого следует, что задача разбиения базы на группы и дальнейшее сопоставление отпечатка нужной группе - актуальная и важная задача. Для её решения могут быть полезны методы машинного обучения, а именно алгоритмы кластеризации данных для разделения базы на группы с похожими свойствами, и методы глубокого обучения, для дальнейшего сопоставления отпечатка в нужный кластер. Под нужным кластером подразумевается тот, в котором есть изображения искомого пальца. В данной работе предполагается применить данные методы для решения задачи увеличения скорости идентификации отпечатков пальцев.

1. Постановка задачи

Целью данной дипломной работы является применение методов глубокого обучения для решения задачи кластеризации базы отпечатков пальцев. В ходе работы были поставлены следующие задачи:

- Разработать систему, разделяющую базу отпечатков пальцев на кластеры и сопоставляющую отпечаток в нужный кластер.
- Сравнить результаты работы системы с существующими решениями задачи идентификации отпечатка пальца в базе

2. Исходные данные

В качестве исходных данных были использованы отпечатки пальцев из базы DB2, которая была представлена на международном соревновании по распознаванию отпечатков пальцев FVC2002 (The Second International Competition for Fingerprint Verification Algorithms). База содержит в себе изображения пальцев 110 человек, по 8 различных изображений одного пальца человека.

Разработанная в рамках данной работы система использовала базу, содержащую по одному изображению пальца человека. Таким образом, она содержала 110 изображений. Такой подход максимально приближен к реальным условиям, т.к. в коммерческих решениях в базах хранится только одно изображение пальца человека.

3. Обзор существующих решений

Первые попытки классификации отпечатков были предприняты еще в 1892 году Гальтоном [3] и в 1900 году Генри [2]. Этот способ получил название классификация по Гальтону-Генри. На рисунке 1 можно увидеть 4 группы, на которые они разделяли отпечатки.



Рис. 1: Классификация отпечатков по Гальтону-Генри

В целом, существует ряд подходов, реализующих автоматическое сопоставление отпечатков заранее определенным классам. В статье [1] представлен метод, в котором вектор признаков получается при помощи фильтра Габора, и классификация осуществляется методом ближайших соседей. В статье [17] также используется фильтр Габора, и классификация осуществляется методом опорных векторов. В [7] наоборот, использует поле направлений, как способ получения вектора признаков, и метод опорных векторов для классификации.

Решение, описанное в [12], использует поле направлений в качестве вектора признаков и классификация происходит некоторыми стохастическими подходами. Также есть ряд работ, использующих нейронные сети [9] [1]

В целом, видна тенденция использования методов машинного обучения, подходы в основном различаются за счет выбора способа получения вектора признаков.

3.1. Метрика сравнения алгоритмов

Для того, чтобы сравнивать алгоритмы, использовалось общепринятое для методов идентификации отпечатков пальцев понятие точность [4]. Оно удовлетворяет следующей формуле:

$$\text{Точность} = \frac{\text{Количество отпечатков, попавших в нужный кластер}}{\text{Количество отпечатков в базе}} * 100\% \quad (1)$$

Под нужным кластером понимается тот, в котором находится изображение искомого пальца. Предполагается, что берется база отпечатков, далее для каждого отпечатка из базы рассматривается еще одно изображение этого пальца и подаётся на вход системе. Если после работы система соотнесла отпечаток кластеру, в котором есть изображение этого пальца, то считается, что отпечаток попал в нужный кластер.

4. Методология

Всю работу системы, разработанной в рамках данной дипломной работы, можно разделить на несколько этапов. На первом этапе происходит подготовка базы изображений отпечатков для дальнейшей обработки. Каждое изображение заменяется на числовой вектор признаков, с которым удобнее работать. Далее база из векторов подвергается кластеризации, затем нейронная сеть глубокого обучения учится сопоставлять отпечатки пальца нужным кластерам.

Для решения задач, поставленных в рамках данной работы, использовалась библиотека CUDA-Fingerprinting для языка C# [15], а также библиотека Scikit-learn [14] для языка Python. Данные технологии были выбраны потому, что позволяют построить очень гибкую систему, составные части которой можно без труда менять, для поиска наиболее оптимальных параметров.

Ниже будут более детально рассмотрены используемые на каждом этапе алгоритмы.

4.1. Вектор признаков

Получение векторов признаков является очень важной составляющей работы нашей системы. Как видно из обзора, существует много различных способов получения вектора из изображения, однако на сегодняшний день не существует универсального и самого эффективного способа.

Существует ряд решений [7], [9], основывающихся на методе поля направлений, показавших хорошие результаты, поэтому его и решено было использовать. Недостатком данного подхода может являться то, что полученный вектор мог получаться очень большим и содержать много ненужных признаков. Для решения этой проблемы было решено применить метод главных компонент, для уменьшения размерности вектора путем удаления ненужных признаков.

Далее представлено более детальное описание использованных подходов.

4.1.1. Поле направлений

Идея данного метода основана на статье [13] и заключается в следующем. Изображение отпечатка делится на блоки фиксированного размера по линиям сетки. Далее в каждом таком блоке с помощью градиентов считается доминирующее направление папиллярных линий отпечатка. Направление θ_d для каждого блока d вычисляется по следующей формуле:

$$\theta_d = \frac{1}{2} \tan^{-1} \left(\frac{\sum_{i=1}^n \sum_{j=1}^n 2G_x(i, j)G_y(i, j)}{\sum_{i=1}^n \sum_{j=1}^n (G_x(i, j)^2 - G_y(i, j)^2)} \right), G_x \neq 0, G_y \neq 0 \quad (2)$$

где n - это размер блока, $G_x(i, j)$ и $G_y(i, j)$ - это значения градиента по оси x и y у пикселя (i, j)



Рис. 2: Поле направлений

Затем полученные значения направлений записываются в итоговый вектор, на основе которого и происходит дальнейшая работа системы. Например, для размера блока 16×16 на изображениях из базы FVC2002 получается вектор размерности 352.

В качестве недостатка данного метода можно отметить то, что полученные значения для отпечатка никак не устойчивы к поворотам пальца.

4.1.2. Метод главных компонент

Метод главных компонент (principal component analysis, PCA) хорошо показал себя в задаче идентификации отпечатка [9], поэтому было решено попробовать применить данный метод в нашей системе.

Метод главных компонент принимает на вход вектор признаков, полученный методом поля направлений, и далее уменьшает размерность вектора с помощью сингулярного разложения, оставляя при этом наиболее значимые признаки в векторе.

4.2. Кластеризация

Кластеризация (обучение без учителя) решает задачу разбиения множества объектов на непересекающиеся кластеры, причем решение о разбиении принимается на основе собственной работы алгоритма, а не размеченных данных, как, например, в задаче классификации (обучение с учителем). Объекты в одном кластере близки друг к другу по заданной метрике, и наоборот, объект разных кластерах далеки друг от друга.

Более формально, то для каждого объекта X_i задается вектор характеристик $X_i = (x_1..x_n)$, где n - размерность пространства характеристик. Можно измерять расстояния между векторам характеристик $d(x_i, x_j)$ на основе выбранной метрики пространства характеристик.

Можно выделить 4 этапа кластеризации:

1. Получение характеристик из объектов
2. Определение метрики
3. Разделение на кластеры
4. Получение результатов

Стоит отметить, что в рамках данной работы объектами кластеризации являются изображения отпечатков пальцев, а вектора характеристик из изображений получаются на основе методов, описанных в предыдущей секции.

Выделяется два основных вида алгоритмов кластеризации: иерархические и неиерархические. Иерархические разделяются на агломеративные (восходящие, объединительные) и дивизивные (нисходящие, разделяющие). Существует большое количество неиерархических алгоритмов, однако можно выделить несколько основных групп: алгоритмы теории графов, статистические алгоритмы.

Иерархические алгоритмы последовательно строят кластеры из уже имеющихся, в первом случае изначально каждый объект находится в отдельном кластере, и далее кластеры объединяются, во втором наоборот, все объекты в одном кластере и далее он разделяется на несколько.

Основная цель неиерархических алгоритмов - это максимизация или минимизация некой целевой функции, которую можно понимать как критерий оценивания кластеризации. Многие из этих алгоритмов будут итеративно назначать объекты различным кластерам, при этом занимаясь поиском некоторого оптимального значения критерия.

В качестве алгоритмов кластеризации было решено использовать по одному виду кластеризации из каждой группы, а именно агломеративную иерархическую кластеризацию и метод k-средних (KMeans). Были выбраны именно эти алгоритмы потому, что они являются самыми популярными методами кластеризации, а также они принимают на вход количество кластеров, на которые будет разделена база. Данное условие очень важно, т.к. в зависимости от размера базы отпечатков нам нужно менять то, на сколько мы будем разделять базу, для увеличения точности и скорости идентификации.

Далее будут рассмотрены два алгоритма кластеризации, использованные в данной работе.

4.2.1. Агломеративная иерархическая кластеризация

Основной идеей агломеративной иерархической кластеризации является то, что она объединяет объекты в кластеры снизу вверх, т.е. сначала все объекты находятся в отдельных кластерах, а далее на каждом этапе работы алгоритма они объединяются в более крупные кластера. На каждом этапе объединяются два самых близких кластера. В дан-

ной работе мера близости кластеров получалась по методу Уорда [6]. Основная его идея заключается в том, что объединяются те кластера, которые минимизируют квадрат расстояний объектов до центров кластеров, если бы их объединили. Расстояния между объектами в данной работе считались как Евклидово расстояние.

4.2.2. Метод k-средних

Основополагающей идеей кластеризации методом k-средних является то, что алгоритм стремится уменьшить суммарное квадратичное отклонение векторов кластеров от центров этих кластеров.

На каждом этапе работы алгоритма пересчитывается центр масс для каждого кластера, полученного на предыдущем этапе, затем векторы разделяются на кластеры снова в соответствии с тем, какой из новых центров оказался ближе по выбранной метрике. В качестве метрики расстояния между векторами была выбрана Евклидова метрика.

4.3. Глубокое обучение

Глубокое обучение нейронных сетей – это одна из областей машинного обучения, позволяющая представлять данные на нескольких уровнях абстракции и позволяющая увеличивать сложность абстракции от уровня к уровню. Под глубиной понимается соответственно глубина графа вычислений от входных данных до выходных.

В рамках данной работы была рассмотрена модель нейронной сети под названием многослойный перцептрон Румельхарта. На рисунке 3 представлена её архитектура с двумя скрытыми слоями.

На вход нейронная сеть получает базу, состоящую из векторов признаков, и метки кластеров для каждого вектора признаков из базы. Далее нейронная сеть обучается, а потом ей на вход подается новое изображение, и она соотносит изображение в один из кластеров.

В данной работе использовалась нейронная сеть с тремя скрытыми слоями, в качестве активационной функции использовалась логистическая функция. Алгоритмом для оптимизации весов являлся алгоритм l-

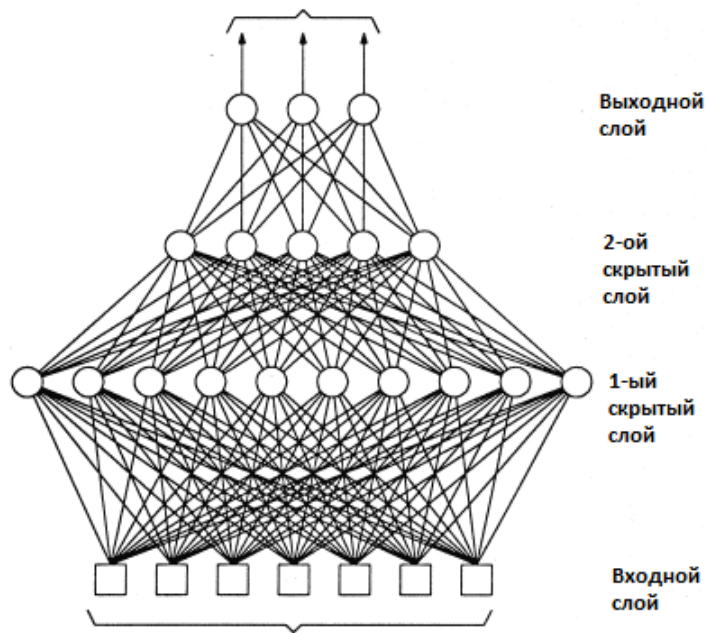


Рис. 3: Архитектура используемой нейронной сети

bfgs - алгоритм оптимизации из семейства квазиньютоновских методов [16]. Для улучшений качества обучения нейросети применялась кросс-валидация, 10% данных разделялось на валидационное множество.

5. Эксперименты

Был произведен ряд экспериментов для выявления самых оптимальных методов получения вектора признаков и самого оптимального способа кластеризовать базу. Оптимальность в данном случае измерялась точностью работы системы.

Эксперименты производились следующим образом. Бралась конкретная конфигурация системы, с конкретным алгоритмом кластеризации и методом получения вектора признаков. Далее нейронная сеть обучалась с различным количеством нейронов в скрытых слоях. Для каждой конфигурации нейронов в скрытых слоях считалась точность соотношения нового изображения каждого пальца из базы нужному кластеру. Например, в нашем случае в базе 110 отпечатков, т.е. 110 различных людей. Далее брались 110 других изображений пальца этих же людей, и считалось, сколько отпечатков попало в кластер, в котором есть изображение искомого пальца, величина точности получалась по формуле 1.

Стоит отметить, что все эксперименты проводились с входным параметром для кластеризации, равным 5, т.е. база разделялась на 5 частей.

Для начала было выяснено, что размер блока 16x16 является наиболее точным. Результаты сравнения точности можно увидеть в таблице 1. В дальнейшем было решено использовать именно размер блока 16x16.

Размеры блока	Точность, %	Кол-во нейронов в слоях
16x16	81,3	57, 52, 25
32x32	67,2	58, 37, 21

Таблица 1: Сравнение точности для размеров блока 16x16 и 32x32

Далее в таблице 2 приведены результаты экспериментов для каждого из методов кластеризации с применением различных способов получения вектора признаков.

Подходы	Точность, %	Кол-во нейронов
Иерарх. + поле направлений	81.3	57, 52, 25
К-средних + поле направлений	85.4	17, 17, 11
Иерарх. + PCA	87.2	38, 34, 21
К-средних + PCA	88.2	86, 29, 29

Таблица 2: Сравнение точности различных подходов кластеризации и различных способов получения вектора признаков

Как видно из таблицы, метода главных компонент (PCA) показал точность выше, чем просто поле направлений, а также кластеризация методом к-средних показывает точность выше, чем агломеративная иерархическая кластеризация.

В итоге, максимальная точность системы равна 88.2%, такая точность получена при использовании метода главных компонент для получения вектора признаков, метода к-средних для кластеризации, и при 86 нейронах в первом, 29 нейронах во втором, и 29 нейронах в третьем скрытых слоях.

6. Сравнение с аналогами

Ниже приведена таблица, сравнивающая точность известных решений данной задачи и точность нашего подхода. В каждом из подходов точность считалась с учётом того, что база разделялась на 5 частей. Данные для точности известных решений получены из статьи [18]

Подходы	Точность, %
Wilson et al. [11]	81.0
Karu et al. [8]	85.4
Jain et al. [1]	90.0
Hong et al. [10]	87.5
Cappelli et al. [12]	92.2
Yao et al. [17]	89.3
Chang et al. [5]	94.8
Zhang et al. [7]	84.0
Наш подход	88.2

Таблица 3: Сравнение существующих подходов и нашего подхода

Из таблицы видно, что наше решение обошло некоторые подходы по точности, однако не показало наилучших результатов. Однако наша точность 88.2% получена всего на 110 примерах отпечатков, в то время как подходы из таблицы тестировались на более крупных выборках, которых нет в открытом доступе. Это очень критично, т.к. в нашем решении имеет место обучение нейронной сети, а её успешная работа зависит от количества примеров в базе, и чем выше это количество, тем точнее должна работать нейронная сеть. Таким образом, при более крупных базах наш подход должен получать еще более точные результаты.

Заключение

В рамках данной работы получены следующие результаты:

- Разработана система, разделяющая базу отпечатков пальцев на кластеры и сопоставляющая отпечаток в нужный кластер.
- Проведено сравнение результатов работы системы с существующими решениями задачи идентификации отпечатка пальца в базе.

Список литературы

- [1] A. Jain, S. Prabhakar, L. Hong. A multichannel approach to fingerprint classification // IEEE Trans Patt Anal Mach Intell. — 1999. — Vol. 21. — P. 348–359.
- [2] E. Henry. Classification and uses of finger prints // Rutledge. — 1900.
- [3] F. Galton. Finger prints // McMillan. — 1892.
- [4] Handbook of fingerprint recognition / Davide Maltoni, Dario Maio, Anil Jain, Salil Prabhakar. — Springer Science & Business Media, 2009.
- [5] J. Chang, K. Fan. A new model for fingerprint classification by ridge distribution sequences // Patt Recog. — 2002. — Vol. 35(6). — P. 1209–1223.
- [6] J.H. Ward. Hierarchical grouping to optimize an objective function // J. of the American Statistical Association. — 1963. — P. 236.
- [7] Ji Luping, Yi Zhang. SVM-based Fingerprint Classification Using Orientation Field // 3rd International conference on Natural Computation. — 2007. — Vol. 2. — P. 724–727.
- [8] K. Karu, A. Jain. Fingerprint classification // Patt Recog. — 1996. — Vol. 29(3). — P. 389–404.
- [9] Kamijo Masayoshi. Classifying Fingerprint Images using Neural Network: Deriving the Classification State // IEEE International Conference on Neural network. — 1993. — Vol. 3. — P. 1932–1937.
- [10] L. Hong, A. Jain. Classification of fingerprint images // Proceedings of the 11th Scandinavian Conference on Image Analysis. — 1999.
- [11] Massively parallel neural network fingerprint classification system / Wilson C., Candela G. abd Grother P., Watson C., Wilkinson R. // National Institute of Standards and Technology. — 1992.

- [12] R. Cappelli, D. Maio, D. Maltoni. Fingerprint classification based on multi-space KL // Proceedings of the Workshop on Automatic Identification Advances Technologies. — 1999.
- [13] Ratha Nalini K, Chen Shaoyun, Jain Anil K. Adaptive flow orientation-based feature extraction in fingerprint images // Pattern Recognition. — 1995. — Vol. 28, no. 11. — P. 1657–1672.
- [14] Scikit-learn: Machine learning in Python / Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort et al. // The Journal of Machine Learning Research. — 2011. — Vol. 12. — P. 2825–2830.
- [15] Stanislav Sartasov. CUDA-Fingerprinting. — URL: <https://github.com/Stanslav-Sartasov/CUDA-Fingerprinting> (online; accessed: 2016-05-19).
- [16] Wikipedia. Limited-memory BFGS // Википедия, свободная энциклопедия. — 2016. — URL: https://en.wikipedia.org/wiki/Limited-memory_BFGS (online; accessed: 08.04.2016).
- [17] Y. Yao, P. Frasconi, M. Pontil. Fingerprint classification with combinations of support vector machines // Proceedings of the 3rd International Conference on Audio and Video Based Biometric Person Authentication. — 2001.
- [18] Yager Neil, Amin Adnan. Fingerprint classification: a review // Springer-Verlag London. — 2004.