УДК 004.934.8'1

И.А. Рахманенко

Программный комплекс для идентификации диктора по голосу с применением параллельных вычислений на центральном и графическом процессорах

Статья посвящена программному комплексу для идентификации диктора по голосу с применением параллельных вычислений на центральном и графическом процессорах. В качестве основы для построения данного комплекса были использованы модели Гауссовых смесей и универсальная фоновая модель (GMM-UBM система).
Разработанный комплекс позволяет производить обучение универсальной фоновой модели (UBM), моделей дикторов и производить тестирование речевых сегментов на принадлежность заданной модели диктора. Комплекс позволяет производить отбор речевых признаков с помощью алгоритмов жадного добавления-удаления и генетического алгоритма.
Произведена экспериментальная оценка скорости работы модуля обучения универсальной фоновой модели в различных реализациях – на центральном процессоре, на процессоре видеокарты и в комбинированном варианте. Реализованный модуль обучения УФМ с комбинированными вычислениями на центральном процессоре и процессоре видеокарты, по сравнению с обучением УФМ на центральном процессоре, позволяет уменьшить время работы на 36,95%, по сравнению с обучением на процессоре видеокарты – на 10%.
Ключевые слова: распознавание диктора, верификация диктора, Гауссовы смеси, GMM-UBM-система, обработка речи, программный комплекс, параллельные вычисления, GPU, CUDA.
doi: 10.21293/1818-0442-2017-20-1-70-74

В настоящее время для науки актуальными являются задачи обработки данных, среди которых можно выделить задачи обработки речи. В данной области одной из сложных и требующих решения задач является задача автоматической идентификации диктора по голосу. Известно много современных систем, которые пытаются решать данную задачу достаточно эффективно, однако точность подобных систем не всегда соответствует достаточному уровню для их реального применения. Кроме того, отдельным вопросом стоит большой объем данных, требующих обработки. Все более и более возрастающий объем данных требует разработки таких решений, которые бы позволили эффективно и быстро производить необходимые вычисления. К таким затратным процедурам в области идентификации диктора по голосу относят обучение универсальной фоновой модели (universal background model, UBM) и моделей дикторов. В разработанном программном комплексе были произведены попытки улучшить как точность распознавания диктора, так и уменьшить время обработки данных.

Задача распознавания диктора включает в себя две подзадачи: идентификацию и верификацию. Автоматическая верификация диктора – это подтверждение личности по голосу в соответствии с предъявленным им идентификатором (обычно именем данного диктора). Отличие же автоматической идентификации диктора заключается в том, что изначально неизвестен идентификатор диктора, соответственно система должна сама определить, кем является данный диктор – законным пользователем, зарегистрированным в системе, или нарушителем (в случае решения задачи открытой идентификации) [1]. Система автоматической текстонезависимой верификации диктора, представленная в данной работе, решает задачу верификации закрытого множества дикторов, решая, присутствует ли на аудиозаписи голос заявленного диктора или нет. В данном случае существование дикторов, не зарегистрированных в системе, не принимается во внимание.

На точность современных систем распознавания диктора накладывается довольно много ограничений. Сюда относят проблемы, связанные с несоответствием условий обучения и распознавания диктора, проблемы различных акустических условий, в том числе наличия посторонних шумов и помех, проблемы отличия в спектральных составляющих записей голоса из-за применения различных микрофонов. Все это, в дополнение к несовершенству моделей и методов, применяемых для идентификации диктора, ведет к уменьшению точности идентификации.

Требования к точности идентификации диктора для подобных систем задают определенную планку, которая повышается с каждым годом. Однако, несмотря на достаточно широкий спектр возможных недостатков систем аутентификации по голосу, нельзя недооценивать их достоинства, благодаря которым они получили свое распространение в таких областях, как системы биометрической многофакторной аутентификации, системы дистанционного банковского обслуживания, системы контроля доступа, и многих других. Применение подобных систем позволяет повысить надежность систем аутентификации и упростить аутентификацию для конечного пользователя, так как для него отпадает необходимость в запоминании паролей. Однако, в интересах же конечного пользователя и высокая надежность подобных систем, так как ему необходима сохранность данных и финансов, поэтому и выдвигаются высокие требования по точности распознавания диктора.

Для оценки точности идентификации диктора используют несколько характеристик, одна из которых является наиболее часто используемой – равная ошибка первого и второго рода (Equal Error Rate, EER). Данная характеристика используется как для оценки текстозависимых, так и текстонезависимых систем идентификации диктора. Лучшие системы идентификации диктора, тестируемые на фиксированной базе данных, содержащей фразы нескольких сотен дикторов, показывают значение EER 3–5% [2], испытания проводятся в Национальном институте стандартов и технологий США (NIST).

Для применения в реальных системах данной точности недостаточно. С одной стороны, наиболее важной можно считать ошибку второго рода, когда за легального пользователя системы принимается самозванец, соответственно можно сместить порог принятия решений системы в сторону уменьшения данной ошибки. Однако это повлечет за собой увеличение ошибок первого рода, т.е. увеличит частоту отказов легальным пользователям на доступ к системе, что может повлечь за собой недовольство пользователей, использующих систему. Следовательно, необходимо направить усилия для улучшения точности методов идентификации диктора по голосу, что позволит снизить вероятность потери конфиденциальной информации в случае применения в реальных системах.

Применяемые модели

В данный момент для идентификации диктора применяется достаточно большое количество различных моделей, одной из которых является Гауссова смесь, используемая в данной работе.

Гауссова смесь (ГС) – это параметрическая функция плотности вероятности, представленная как взвешенная сумма отдельных Гауссовых плотностей [3]. ГС, состоящая из C плотностей вероятности, может быть представлена формулой:

$$p(x|\lambda)=\sum_{i=1}^{C} w_i g(x|\mu_i,\Sigma_i), \qquad (1)$$

где x – D-мерный непрерывный вектор данных (признаков); w_i, $i=1,...,C$ – вес i-го компонента смеси, и $g(x|\mu_i,\Sigma_i)$; $i=1,...,C$ – Гауссова плотность вероятности i-го компонента смеси с вектором математических ожиданий μ_i и ковариационной матрицей Σ_i. Таким образом, полную ГС можно описать множеством векторов математического ожидания, ковариационных матриц и весов смесей каждого компонента модели. ГС можно представить уравнением

$$\lambda=\{w_i,\mu_i,\Sigma_i\}. \qquad (2)$$

Итого при решении задачи распознавания диктора каждый из дикторов представлен в системе собственной ГС λ.

Гауссовы смеси используют в задачах идентификации диктора благодаря двум наблюдениям [4]. Во-первых, было замечено, что индивидуальные компоненты смеси могут моделировать некоторое множество акустических классов. Данное множество представляет собой набор конфигураций голосового тракта диктора, что позволяет использовать их в целях идентификации. Акустические классы являются «скрытыми», так как в обучающих и контрольных данных они не размечены. Если предположить, что векторы признаков независимы друг от друга, то Гауссова смесь описывает эти классы через плотность распределения наблюдаемых векторов признаков

Во-вторых, линейная комбинация нормальных распределений может представлять большое множество распределений акустических признаков. Достоинством Гауссовой смеси является способность точной аппроксимации распределений произвольной формы. Можно сказать, что Гауссова смесь представляет собой нечто среднее между методом векторного квантования, где распределение признаков представлено дискретным множеством шаблонов, и одним Гауссовым распределением с единственным вектором математических ожиданий и ковариационной матрицей.

Универсальная фоновая модель (UBM, УФМ) – это ГС, обученная на большом наборе речевого материала, взятого от большого множества дикторов, ожидаемых системой во время распознавания. Благодаря этому можно использовать УФМ для проверки альтернативной гипотезы, т.е. того случая, когда на записи отсутствует голос заданного диктора. Как и в [5], параметры для УФМ были обучены с помощью EM-алгоритма, а для обучения моделей дикторов была использована форма Байесовой адаптации.

Для существующих систем идентификации используются базы речевых данных в несколько сотен часов. При этом обучение УФМ может длиться не одну неделю на современном центральном процессоре, а существенное увеличение размера базы становится практически невозможным [6]. Для ускорения процесса обучения УФМ можно использовать параллельные алгоритмы, в том числе с применением вычислений на графическом процессоре видеокарты.

Для обучения модели диктора и УФМ λ наиболее часто используют метод максимального правдоподобия. Данный метод позволяет подобрать параметры модели по обучающим данным таким образом, чтобы функция правдоподобия модели достигла максимума.

Для последовательности из T обучающих векторов $X=\{x_1,x_2,...,x_T\}$, функция правдоподобия может быть записана как [5]

$$p(X|\lambda)=\prod_{t=1}^{T} p(x_t|\lambda). \qquad (3)$$

Напрямую максимизировать (3) невозможно, однако приближенное значение можно с помощью алгоритма EM (expectation-maximization). С помощью данного алгоритма вычисляется ожидаемое значение функции правдоподобия (4), после чего находят оценку максимального правдоподобия для каждой компоненты модели и вычисляют новые компоненты модели [4].

$$\Pr(i|x_t) = w_i p_i / \sum_{j=1}^{C} w_j p_j(x_t), \quad (4)$$

$$w = \left(\sum_{t=1}^{T} \Pr(i|x_t) \right) / T, \quad (5)$$

$$\mu = \left(\sum_{t=1}^{T} \Pr(i|x_t) x_t \right) / \sum_{t=1}^{T} \Pr(i|x_t), \quad (6)$$

$$\Sigma = \left(\sum_{t=1}^{T} \Pr(i|x_t) x_t^2 \right) / \sum_{t=1}^{T} \Pr(i|x_t) - \mu^2. \quad (7)$$

Была использована ГС, состоящая из 256 компонентов, так как было замечено, что EER не уменьшался при увеличении компонент смеси. Модели дикторов были получены с помощью MAP адаптации с адаптацией только векторов математических ожиданий с фактором релевантности $r = 10$.

Описание программного комплекса

Разработанный программный комплекс предназначен для проведения автоматической верификации или идентификации диктора, при этом включает в себя все необходимые модули для извлечения речевых признаков, обучения моделей дикторов и УФМ, а также проведения верификационных испытаний. GMM-UBM система, описанная в данном разделе, была создана с применением библиотеки MSR Identity Toolbox [7].

Рассмотрим структуру программного комплекса (рис. 1).

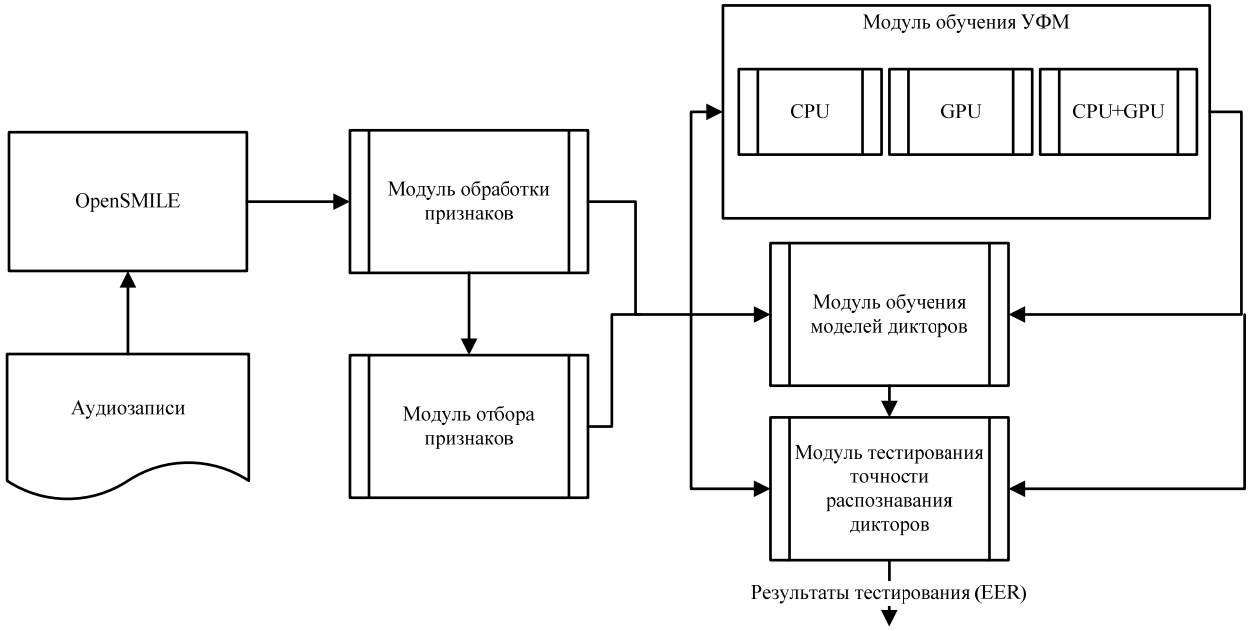

Рис. 1. Структура программного комплекса

Для извлечения речевых признаков из аудиозаписей голоса диктора был использована библиотека openSMILE [8]. Из аудиозаписей были извлечены такие признаки, как мел-кепстральные коэффициенты, пары линейного спектра, кепстральные коэффициенты перцептивного линейного предсказания, кратковременная энергия, формантные частоты, частота основного тона, вероятность вокализации, частота пересечения нуля, джиттер и шиммер. Полный вектор признаков, вычисляемый для одного окна длиной в 20 мс, состоит из 94 признаков.

Для проведения экспериментов было создано несколько модулей – модуль обработки признаков, который позволяет отобрать необходимые в исследовании речевые признаки, модуль обучения УФМ, модуль обучения моделей дикторов и модуль тестирования точности распознавания дикторов. Модуль обучения УФМ был реализован в нескольких вариациях – с вычислениями на центральном процессоре, с вычислениями на процессоре видеокарты и комбинированный вариант.

Для проведения серии экспериментов по отбору речевых признаков были разработаны модули, реализующие алгоритм жадного добавления-удаления и генетический алгоритм. Отбор признаков позволяет снизить переобучение модели и сохранить при этом наиболее информативные признаки.

Жадный алгоритм добавления-удаления признаков [9] включает в себя две жадные стратегии, т.е. производится поочередное добавление и удаление признаков из текущего множества. Сначала алгоритм добавления Add последовательно добавляет признаки до тех пор, пока не начнет увеличиваться ошибка EER и еще $d = 3$ шагов с увеличением ошибки. После этого начинает работу алгоритм жадного удаления Del, который удаляет избыточные признаки.

Генетический алгоритм [10] осуществляет поиск наилучшего набора признаков с использованием методов естественной эволюции. Случайным образом формируется несколько наборов признаков, называемых индивидами, которые объединяются в популяцию. К полученным индивидам случайным

образом применяются операции мутации и скрещивания (кроссовера), таким образом получая новые индивиды. В конце каждой итерации генетического алгоритма производится отбор лучших индивидов, для которых значение целевой функции (в данном случае, EER) является наилучшим.

Одной из ключевых особенностей реализованного программного комплекса является предложенный алгоритм комбинированных вычислений на центральном процессоре и процессоре видеокарты. При этом в аналогах производятся вычисления либо только на центральном процессоре [7, 11], либо только на процессоре видеокарты [6].

Наиболее трудоемким по количеству затрачиваемого времени и вычислений является обучение УФМ, однако обучение данной модели хорошо распараллеливается. Это возможно благодаря разделению последовательности входных обучающих векторов на отдельные блоки, каждый из которых вычисляется отдельно (3), (4), а затем суммируется. При этом полученная сумма не изменится, как и в случае если входные данные на блоки не разбивались бы. То есть возможно выполнение расчетов отдельных частей в разных потоках на разных данных, соответственно каждый из этих потоков независим и работает со своими данными.

Для выполнения одновременных вычислений на центральном (ЦП) и графическом (ГП) процессорах часть блоков данных перемещается в память видеокарты, а затем в отдельном потоке запускаются необходимые вычисления. Для хранения в памяти параметров модели Σ, μ, и w требуется $8*C*D$ байт, в данном случае количество компонент смеси $C = 256$, количество используемых признаков $D = 28$, итого 57344 байт. Для вычислений используются блоки данных размером 50000 векторов по $8*D$ байт, итого $11{,}2\times10^6$ байт. Для промежуточных вычислений необходимы блоки $2*8*C*D$ байт, $8*50000*C$ байт, итого $\approx 102{,}5\times10^6$ байт.

Результаты испытаний программного комплекса

Проведены эксперименты с применением речевого корпуса, включающего записи речи 25 дикторов-мужчин и 25 женщин. Данный речевой корпус содержит записи произнесенных без предварительной подготовки предложений, взятых из художественной литературы, или поговорок. Суммарная длина записей речи для каждого диктора составляет не менее 6 мин, включая 50 сегментов различной длины. Каждый диктор был записан на микрофон в условиях небольшого шума, частота дискретизации 8000 Гц, разрядность 16 бит.

Весь речевой корпус, состоящий из записей речи 50 дикторов, был разделен на обучающую выборку для УФМ, состоящую из записей 30 дикторов, и выборку, использующуюся для обучения и тестирования моделей дикторов, состоящую из записей оставшихся 20 дикторов. Все выборки были выполнены с равным разделением по дикторам разного пола. Общий объем данных, используемых для обучения UBM, составляет 162,28 Мб.

Эксперименты проводились с использованием 4-ядерного процессора Intel Core i7-3630QM, видеокарты nVidia GeForce GT 640M с 2 Гб DVRAM.

Для сравнения эффективности работы разработанного программного комплекса, а конкретно модуля обучения УФМ, было проведено несколько экспериментов по определению скорости работы (времени исполнения) модулей. Сравним время обучения УФМ с разбиением на различные размеры блоков (таблица). При этом зафиксируем количество используемых потоков процессора, равное 4.

Время обучения УФМ в зависимости от размера блоков обучающих данных

Размер блока данных (семплов)	Время обучения на ЦП, с	Время обучения на ГП, с	Время обучения на ЦП и ГП, с
5000	168,7424	165,7454	135,4970
10000	165,5472	128,2411	107,9179
25000	162,0763	116,2236	104,0332
50000	159,0245	111,4273	100,2596

Наименьшее время обучения УФМ было получено при использовании параллельных вычислений на центральном процессоре и видеокарте, которое составляет 100,2596 с.

Сравним скорость вычислений в зависимости от количества запущенных потоков процессора, зафиксировав размер блоков обучающих данных на 50000 векторов (рис. 2). Можно отметить, что независимо от количества потоков комбинированные вычисления на процессоре и видеокарте быстрее, чем только на центральном процессоре. При использовании более 4 потоков и комбинированных вычислений на ЦП и ГП время вычислений далее не уменьшается, при вычислениях на ЦП – уменьшается, но незначительно.

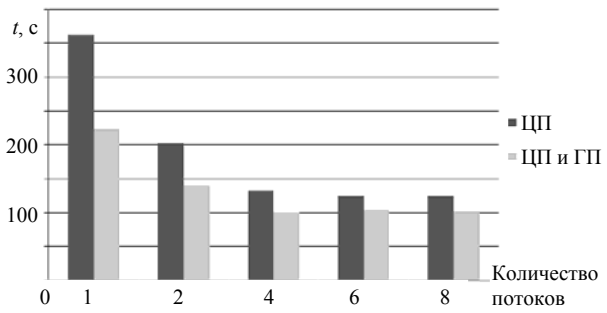

Рис. 2. Время обучения УФМ в зависимости от количества используемых потоков

Кроме того, можно добиться больших результатов, учитывая особенности работы с памятью DVRAM видеокарты, а также с помощью минимизации обращений к памяти и объединения нескольких запросов в один.

Выводы

Был разработан программный комплекс для верификации диктора, основанный на модели Гауссовых смесей и универсальной фоновой модели. Дан-

ный комплекс позволяет производить обучение УФМ, обучение моделей дикторов и производить тестирование речевых сегментов на принадлежность заданной модели диктора. Кроме того, данный комплекс позволяет произвести отбор речевых признаков с помощью алгоритма жадного добавления-удаления и генетического алгоритма.

Были произведены эксперименты по определению скорости работы модуля обучения УФМ в различных реализациях – на центральном процессоре, на процессоре видеокарты и комбинированный вариант. Реализованный модуль обучения УФМ с комбинированными вычислениями на центральном процессоре и процессоре видеокарты по сравнению с обучением УФМ на центральном процессоре позволяет уменьшить время работы на 36,95%, по сравнению с обучением на процессоре видеокарты позволяет уменьшить время работы на 10%.

Литература

1. Campbell Jr.J.P. Speaker recognition: a tutorial // Proceedings of the IEEE. – 1997. – Vol. 85, No. 9. – PP. 1437–1462.

2. Сорокин В.Н. Распознавание личности по голосу: аналитический обзор / В.Н. Сорокин, В.В. Вьюгин, А.А. Тананыкин // Информационные процессы. – 2012. – Т. 12, вып. 1. – С. 1–30.

3. Reynolds D.A. Gaussian mixture models // Encyclopedia of biometric recognition. – Heidelberg: Springer, 2015. – PP. 827–832.

4. Reynolds D.A. Robust text-independent speaker identification using Gaussian mixture speaker models / D.A. Reynolds, R.C. Rose // IEEE Transactions on Speech and Audio Processing. – 1995. – Vol. 3, No. 1. – PP. 72–83.

5. Reynolds D.A. Speaker verification using adapted Gaussian mixture models / D.A. Reynolds, T.F. Quatieri, R.B. Dunn // Digital Signal Processing. – 2000. – Vol. 10, No. 1. – PP. 19–41.

6. Габдуллин В.В. Применение технологии CUDA для задач голосовой биометрии на примере построения универсальной фоновой модели диктора / В.В. Габдуллин, А.И. Капустин, А.И. Королев // Параллельные вычислительные технологии (ПаВТ'2011): труды международной научной конференции. – Челябинск: Изд. центр ЮУрГУ. – 2011. – С. 107–116.

7. Sadjadi S.O. MSR identity toolbox v1.0: A MATLAB toolbox for speaker-recognition research / S.O. Sadjadi, M. Slaney, L. Heck // Speech and Language Processing Technical Committee Newsletter. – 2013. – Vol. 1, No. 4. – PP. 1–32.

8. Eyben F. Recent developments in opensmile, the munich open-source multimedia feature extractor / F. Eyben, F. Weninger, F. Gross, B. Schuller // Proceedings of the 21st ACM international conference on Multimedia. – 2013. – PP. 835–838.

9. Кормен Т. и др. Алгоритмы. Построение и анализ. Гл. 16. Жадные алгоритмы / пер. с англ. – М.: Вильямс, 2012. – 1296 с.

10. Holland J.H. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. – MIT press, 1992. – 232 p.

11. Bonastre J.F. ALIZE, a free toolkit for speaker recognition / J.F. Bonastre, F. Wils, S. Meignier // Acoustics, Speech, and Signal Processing. – 2005. – Vol. 1. – PP. 737–740.

Рахманенко Иван Андреевич
Ассистент каф. безопасности информационных систем (БИС) ТУСУРа
Тел.: +7 (382-2) 70-15-29
Эл. почта: ria@keva.tusur.ru

Rakhmanenko I.A.

Software system for speaker verification using parallel CPU and GPU computing

This paper is devoted to speaker verification software using parallel CPU and GPU computing. This software is based on Gaussian mixture model and universal background model (GMM-UBM system).

Developed software allows to train the universal background model (UBM), speaker models and test recorded speech samples in order to verify their belonging to the selected speaker model. Also, software provides speech feature selection module using greedy add-del and genetic algorithms.

The experimental evaluation of the UBM training module was conducted using CPU, GPU and combined parallel calculations. Parallel CPU and GPU calculations results in 36,95% calculations time decrease compared to parallel CPU implementation, and 10% decrease compared to only GPU implementation.

Keywords: speaker recognition, speaker verification, Gaussian mixture model, GMM-UBM system, speech processing, software system, parallel computations, GPU, CUDA.