



Литература

1. Ojala, T., Pietikainen, M. & Harwood, D. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. Pattern Recognition, 1994. Vol. 1—Conference A: Computer Vision and Image Processing., Proceedings of the 12th IAPR International Conference on 1, 582–585 (1994)
2. Dyck, D. V. Wavelets for texture analysis, an overview. IET Conference Proceedings 581–585 (1997)
3. Haralick R. M., Shanmugam K. & Dinstein I. Textural Features for Image Classification. IEEE Transactions on Systems, Man, and Cybernetics 3, 610–621 (1973)
4. Raheja J. L., Kumar S. & Chaudhary A. Fabric defect detection based on GLCM and Gabor filter: A comparison. Optik—International Journal for Light and Electron Optics 124, 6469–6474 (2013)
5. Hovland, C. I. Computer simulation of thinking. American Psychologist, 15(11), 687-693 (1960).
6. VijayaLakshmi B. & Mohan V. Kernel-based PSO and FRVM: An automatic plant leaf type detection using texture, shape, and color features. Computers and Electronics in Agriculture 125, 99–112 (2016)
7. Brynolfsson, P., Nilsson, D., Torheim, T. et al. Haralick texture features from apparent diffusion coefficient (ADC) MRI images depend on imaging and pre-processing parameters. Scientific Reports 7, Article number: 4041 (2017)
8. Löfstedt T, Brynolfsson P, Asklund T, Nyholm T, Garpebring A Gray-level invariant Haralick texture features. PLoS ONE 14(2) (2019)

В.А. Егунов, С. В. Панюлайтис

РАСПОЗНАВАНИЕ РЕЧЕВЫХ КОМАНД С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ НА НИЗКОПРОИЗВОДИТЕЛЬНЫХ КОМПЬЮТЕРАХ

(Волгоградский государственный технический университет)

Введение

С каждым годом технологии развиваются все стремительнее. Результатом этого бурного технологического прогресса становится появление множества разнообразных интеллектуальных систем. Мобильные и легкие компьютерные детали становятся более и более интегрированными в повседневную жизнь. Умные часы и очки дополненной реальности могут быть полезны и уже стали широкодоступны. Взаимодействие с такими устройствами отличается от взаимодействия со смартфонами, которые уже приобрели функциональные возможности обычных компьютеров.

Все большую важность приобретает проблема взаимодействия человека и компьютера, увеличивается потребность в поиске новых способов обмена



информацией[1]. Голосовое управление уже давно развивается в этом направлении, и хорошо зарекомендовало себя в различных областях вычислительной техники.

В данной работе предлагается решение, позволяющее повысить удобство и доступность управления низкопроизводительными электронными устройствами.

Распознавание речевых команд включает: запись звука, выделение сигнала, извлечение признаков, классификацию и интерпретацию.

Концептуальное проектирование.

Функционально систему распознавания голосовых команд можно разделить по функциональному назначению на модули, перечисленные ниже:

- модуль выделения сигнала – обрезает ранее созданную аудиозапись, оставляя только ту часть, в которую попадает произнесенное слово или несколько слов;

- модуль извлечения признаков – извлекает из сырых данных признаки, характеризующие команду, для распознавания;

- модуль классификации – отвечает за определение команды по признакам одним из методов; к задаче классификации, которую решает этот модуль, предъявляются высокие требования по точности определения команд и скорости выполнения операции;

- модуль интерпретации – преобразует распознанную команду в текст согласно словарю или в управляющие сигналы для какого-либо устройства.

Используемый язык программирования

В данной работе разрабатывается портативная система, обладающая высокой универсальностью. В качестве языка был выбран C++ из-за его кроссплатформенности и высокой производительности.

Запись звука

Для эффективного распознавания команд нейросетью, звуковой сигнал должен обладать постоянством по частоте, громкости, уровню шума и продолжительности. Исходя из этого, необходимо ввести ограничение на расстояние между диктором и микрофоном, что исключает расположение микрофона на работе[2]. Возможно использование, как выносного микрофона, так и встроенного.

Передача сырых данных

Звуковые данные решено было передавать между программными модулями в виде файлов формата *.wav. Такой метод обусловлен тем, что в проекте используется несколько языков программирования и передача данных в ином виде представляется затруднительной.

Для считывания звука с микрофона был написан bash скрипт, записывающий 2-х секундный монофонический файл a.wav с частотой дискретизации 16000Гц и разрядностью или квантованием(-f cd) в 16 бит.



Обрезка, выделение признаков, классификация

Для выделения слова из 2-х секундного файла составлена программа на языке с++. Алгоритм работы программы можно описать следующим образом:

- определяется максимальная и минимальная амплитуда осциллограммы (для разрядности 16 бит максимальные и минимальные значения не могут превышать +32 767 и - 32 768 соответственно);
- выполняется нормализация амплитуд (деление всех положительных амплитуд на максимальную амплитуду, отрицательных - на минимальную);
- разбивается осциллограмма на 200 участков (подобрано опытным путем);
- для каждого участка определяется среднее положительное значение амплитуды;
- выполняется "прогон" по участкам осциллограммы слева и справа; если значение средней положительной амплитуды больше заданной величины (5% из опыта учета внешних шумов), то предполагается, что начало и конец слова достигнут;
- запись выделенного слова в файл с.wav;
- для выделения признаков каждого слова используется спектральный анализ сигнала с использованием дискретного преобразования Фурье;
- предполагается, что для описания слова в каждом из 40-а интервалов достаточно вычислить значения 8-ми векторов (каждый задается парой значений – частотой и амплитудой); таким образом, слово однозначно для небольшой базы словарного запаса описывается массивом из 640 чисел; подтверждение данного утверждения выполнялось экспериментально путем восстановления осциллограммы слова – команды из полученной спектрограммы функциями синуса по 8-и амплитудам и частотам;
- результатом работы программы new.c является формирование массива из 640 чисел (320 векторов), определяющих произнесенное слово.

Нейронная сеть

Опыт показал, что в качестве сети возможно использование классической полносвязной многослойной искусственной нейронной сети, основанной на алгоритме обучения с обратным распространением ошибок.

Количество нейронов входного слоя должно быть равно числу элементов массива, характеризующего слово-команду. По условию задачи структура должна иметь 640 нейронов во входном слое, 7 нейронов в выходном. Опытным путем установлено, что сеть должна иметь 2 скрытых слоя по 100 нейронов.

Команды

Была создана полная обучающая база команд, состоящая из 336 аудиофайлов, по несколько вариантов произношения на каждое слово.



Прототип системы включает в себя:

- Модуль `slice` – формирование значащего участка из звукового файла.
- Модуль `new` – выделение признаков из звукового файла.
- Обучающий модуль `trainall.php`.
- Программа, использующая обученную нейронную сеть для распознавания команд `test.php`.
- Bash скрипт, последовательно запускающий необходимые программы.

Распознавание

Обученная нейронная сеть позволяет убедиться в работоспособности всей системы. Распознавание команд производится практически без задержек, что подтверждает заложенные требования к производительности.

Заключение

1. Показана возможность надежного распознавания слов с помощью ДПФ интервалов слов длиной 15...23мс с выделением 8-ми локальных максимумов амплитуд. Восстановление слова с помощью функций синуса показало близкое сходство звука исходному слову.

2. С помощью созданной программы нейронной сети показана практическая возможность построения нейронной сети распознавания 7-и команд.

3. Представленный здесь подход практического решения задачи распознавания команд можно использовать для распознавания нескольких десятков команд.

4. Представленное здесь решение можно использовать для создания систем управления голосом на любом языке, любыми звуками.

К недостаткам, замеченным в процессе тестирования системы, можно отнести зависимость качества распознавания от диктора, микрофона и даже расположения микрофона относительно диктора. Для решения этих проблем необходимо создание большей базы данных слов, произнесенных различными дикторами с разными микрофонами и при различных расположениях микрофонов.

Для надежного (практически 100% при правильном произношении слова) распознавания слов необходимо отсутствие посторонних звуков во время произношения слова. В основном посторонние шумы влияют на выделение слова из 2-х секундного аудио файла программой `slice.c`. Можно предположить, что если выделение слова при наличии шумов будет правильным, то нейронная сеть для различных вариантов слов с шумами сможет правильно выполнить распознавание (аналогия распознавания человеком слов с учетом помех).

В ходе работы были выполнены все поставленные задачи:

- проведен анализ существующих решений;
- разработан алгоритм решения задачи распознавания речевых команд;
- разработана программная архитектура проекта;



- создан программный пакет для организации человеко-машинного интерфейса.

Полученная система распознавания в будущем может использоваться во многих проектах требующих голосового управления. Преимуществами разработанной системы над уже существующими программными продуктами являются поддержка любого языка, минимальные требования к ресурсам вычислительной техники, высокая скорость работы, простота использования.

Литература

1 Ульман Л. Основы программирования на РНР : монография / Л. Ульман. . – Москва : ДМК Пресс, 2001. – 288 с. : ил. (Самоучитель).

2 Мазуркевич А. МВ РНР: настольная книга программиста /А. Мазуркевич, Д. Еловой. – Минск : Новое знание, 2003. – 480 с. : ил.

А.Н. Ионов, Л.С. Зеленко, Д.С. Оплачко

РАЗРАБОТКА КЛАССИФИКАТОРА ИЗОБРАЖЕНИЙ С ПОМОЩЬЮ ДЕСКРИПТОРОВ SURF И МЕТОДА ОПОРНЫХ ВЕКТОРОВ

(Самарский университет)

Введение

В современном мире цифровых технологий каждый день появляется достаточно большое количество фотографий. Американская статистика свидетельствует о появлении 9,5 миллиардов фотографий в день, сделанных только американцами, при этом больше трети фотографий снимается телефоном [1]. При большом объеме фотографий поиск нужной фотографии может занять достаточно большое время, кроме того, может возникнуть проблема, связанная с неопределенностью того, что изображено на фотографии.

В связи с этим возникает потребность в автоматизации данного процесса, это можно сделать с помощью SURF-дескрипторов (SURF – Speeded Up Robust Features), которые позволяют выделить основные признаки изображения, и метода опорных векторов (SVM – Support Vector Machines), который позволяет определять принадлежность изображения к тому или иному классу.

Общий принцип работы алгоритма поиска изображения

Методы поиска по содержанию основаны на представлении изображения в виде векторов признаков (дескрипторов) – наборов численных параметров, описывающих характеристики изображения, например, такие как цвет, текстуру и т.д. Вектора признаков принимают значения в пространстве признаков. Если на таком пространстве задать меру, то можно сравнивать изображения друг с другом, вычисляя расстояние между соответствующими векторами признаков.