

сигнала (фальстарт). Более подробный алгоритм работы и длительность засекаемых временных интервалов зависит от выбранного режима. Брейн-система поддерживает проведение двух игр – «Эрудит-квартет» или «Брейн-ринг».

Перечислим главные особенности игр «Эрудит-квартет» или «Брейн-ринг», которые учтены в брейн-системе. В игре «Эрудит-квартет» участвуют четыре команды, общее время для запросов ответа является постоянным вне зависимости от истории нажатия кнопок, однако игрок, который дал неверный ответ не может ответить повторно. Отличия «Брейн-ринга» от «Эрудит-квартет» является участие только двух команд и изменение данного на запрос ответа времени после неправильного ответа одной из команд. [1]

Брейн-система строится на основе микроконтроллера, к которому подключаются динамики для воспроизведения звука, кнопки для взаимодействия с игроками и Bluetooth-модуль для передачи информации от ведущего с помощью мобильного приложения. Отсчет времени, которое дается на ответ, также производит микроконтроллер. Для отображения позиции приоритетного игрока используются светодиоды, которые загораются, когда игроку дается право на ответ. [2]

Для взаимодействия ведущего с брейн-системой используется мобильное приложение для ОС Android. Для обеспечения обмена информацией между микроконтроллером и приложением используется Bluetooth-модуль, который для упрощения установки соединения работает как в режиме ведомого устройства, так и ведущего. [1] Мобильное приложение служит для выбора ведущим режима работы, то есть вида игры, фиксирования правильности ответа и последующей отправки этих данных на микроконтроллер, который использует эту информацию для выбора дальнейших действий, например, для отсчитывания временных интервалов и воспроизведения звуковых сигналов, в соответствии с ходом игры по регламентированным правилам. [2]

Одной из главных целей при создании брейн-системы было модифицировать уже существующие подобные системы, а также учесть недостатки этих систем, которые создают неудобства при работе с ними. Перечислим недостатки существующих брейн-систем:

- 1) Необходимость подключения к компьютеру, так как система взаимодействует со специальным приложением для Windows;
- 2) Отсутствие портативности (во многом это является следствием пункта 1);
- 3) Необходимость засечки времени вручную на дополнительных устройствах, а также нажатия кнопки сброса по истечении времени;
- 4) Отсутствие фиксирования результатов, что увеличивает вероятность появления ошибок при конечном фиксировании результатов.

Главным отличием данной системы от существующих является автоматическое фиксирование результатов, а также отсутствие необходимости отсчета времени вручную с помощью дополнительных средств, что существенно облегчает работу ведущего и сводит к минимуму ошибки при фиксировании результатов игры. Также система обладает большей портативностью, так как использует приложение для смартфонов. Данные модификации делают брейн-систему более удобной и простой в использовании.

Список использованных источников:

1. Белорусская лига интеллектуальных команд [Электронный ресурс]. – Режим доступа: <http://www.blik.by/>. – Дата доступа: 04.04.2019.
2. Arduino [Электронный ресурс]. – Режим доступа: <https://www.arduino.cc/>. – Дата доступа: 05.04.2019.

СЖАТИЕ ИЗОБРАЖЕНИЙ НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ

Никитин Г.Ю.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Петровский Н.А. – к.т.н., доцент

В докладе приводится способ сжатия изображения на основе свёрточных нейронных сетей и нелинейных функций активации.

Сжатие данных и, в частности, изображений - фундаментальная и хорошо изученная проблема в технике. Существует множество подходов, позволяющих решить её, в той или иной степени. Как правило, для компрессии изображений, применяют алгоритмы одной из трёх групп, в зависимости от представления коэффициентов: 1) частотной – дискретное косинусное преобразование (DCT); 2) частотно-временной – на основе вейвлет-анализа (Wavelet) и 3) временной, к примеру RLE. В последнее время для сжатия данных (и изображений) широко применяются и нейронные сети в различных их вариациях. Например, для сжатия изображений используют сеть Кохонена,

ассоциативную память, а также методы «бутылочного горлышка».

Рассмотрим структуру нейронной сети по принципу «бутылочного горлышка», предложенной в работе [1]. Структура слоёв нейронной сети представлена на рисунке 1, а формулы преобразования GDN и IGDN представлены на формулах 1 и 2.

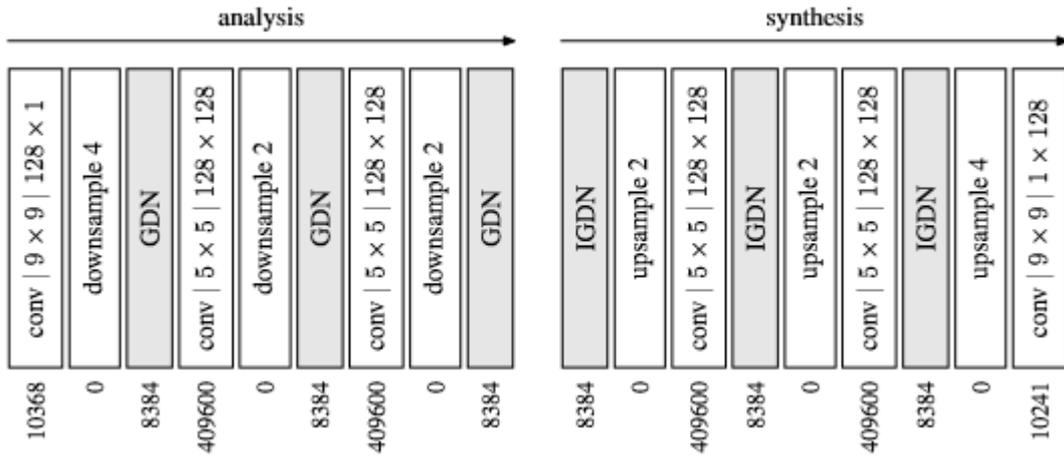


Рисунок 1. – Общая структура нейронной сети.

$$u_i^{(k+1)}(m, n) = \frac{\omega_i^k(m, n)}{\sqrt{\beta_{k,i} + \sum_j \gamma_{k,ij}(\omega_j^k)^2}} \quad (1)$$

Где β и γ – тренируемые параметры;
 ω – набор входных каналов;
 u – набор выходных каналов.

$$\hat{\omega}_i^k(m, n) = \hat{u}_i^k(m, n) * \sqrt{\hat{\beta}_{k,i} + \sum_j \hat{\gamma}_{k,ij}(\hat{u}_j^k(m, n))^2} \quad (2)$$

Где $\hat{\beta}$ и $\hat{\gamma}$ - тренируемые параметры;
 $\hat{\omega}$ - набор выходных каналов;
 \hat{u} - набор входных каналов.

Функция потерь подробно расписана и объяснена в [1].

Модель описана на языке Python при помощи библиотеки Tensorflow и обучена на различных выборках изображений. Ниже, на рисунках 2а и 2б, представлены изображение до сжатия и после сжатия соответственно. Как можно убедиться, с точки зрения перцептуальной приемлемости – различия найти достаточно сложно. В таблице 1 приведены некоторые независимые оценки для нескольких обработанных изображений.



Рисунок 2(а, б). – Изображения до и после сжатия.

Таблица 1. Независимые оценки сжатия.

Тестовое изображение	MSE	PSNR (дБ)	MS-SSIM [2]	MS-SSIM (дБ)	Actual bits per pixel	Information content in bpp
Lena	99.5555	28.15	0.9674	14.87	1.7239	1.7226
Barbara	165.3647	25.95	0.9692	15.12	1.7292	1.7279
Baboon	332.9701	22.91	0.9568	13.64	1.7158	1.7144

Как видно из таблицы 1, оценка «попиксельного» сравнения PSNR даёт достаточно высокие результаты, а в то же время оценка MS-SSIM говорит о более чем 90% совпадении изображений до и после сжатия. Это объясняется тем, что нейронная сеть не стремится сохранить значения «пиксель в пиксель», а стремится сохранить значения у группы пикселей, квантовать их.

В заключение хотелось бы отметить, что представленную модель с большой долей вероятности можно улучшить, предварительно обрабатывая изображения до сжатия. Взглянем на спектральную область изображений. На рисунке 3. спектр изображений до и после сжатия представлен в качестве гистограммы спектральной области изображений в логарифмической системе счисления (дБ).

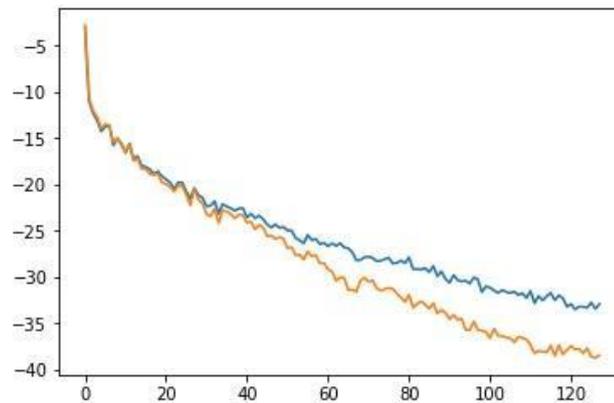


Рисунок 3. – Гистограмма спектра изображения до (верхний график) и после (нижний график) сжатия.

На рисунке 3 можно увидеть подавление высокочастотной части изображения, объясняющееся тем, что в этой частотной области изображения сосредоточены признаки, которые нейронная сеть считает наиболее маловажными для человеческого восприятия. Этот факт позволит оптимизировать входной набор данных и уменьшить нагрузку на нейронную сеть. В качестве такого преобразования можно использовать пакетное дискретное вейвлет преобразование (ПДВП) [3] с различными вариантами вейвлетов и разным построением дерева преобразований.

Список использованных источников:

1. End-to-end optimized image compression [Электронный ресурс] – Режим доступа : <https://arxiv.org/pdf/1611.01704.pdf>
2. Multi-scale structural similarity for image quality assessment [Электронный ресурс] – Режим доступа : <https://ece.uwaterloo.ca/~z70wang/publications/msssim.pdf>
3. A Wavelet Tour of Signal Processing [Электронный ресурс] – Режим доступа : <https://www.di.ens.fr/~mallat/papiers/WaveletTourChap1-2-3.pdf>