МИНИСТЕРСТВО ПО РАЗВИТИЮ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
И КОММУНИКАЦИЙ РЕСПУБЛИКИ УЗБЕКИСТАН
ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

На правах рукописи
УДК 621.391.25

КИМ МАРИНА ВАЛЕНТИНОВНА

Исследование перспективных методов сжатия ТВ изображений на основе нейротехнологий и оценка их эффективности

5А350101 – Системы телевидения и радиовещания

Диссертация на соискание академической степени магистра

Научный руководитель
к.т.н., доцент
Гаврилов И.А.

Ташкент-2015

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

С развитием цифрового телевидения значительно возрастают потребности в увеличении качества и количества ТВ программ, при этом растет количество программ передаваемых в стандартах высокой четкости, имеющих гораздо больший объем данных. При этом в условиях ограниченного частотного ресурса увеличить число передаваемых программ при сохранении качества изображений можно, только создавая более эффективные методы сжатия ТВ изображений, сохраняющие хорошее визуальное качество изображений при больших коэффициентах сжатия видеопотока.

Президент Республики Узбекистан Ислам Абдуганиевич Каримов в своём докладе на заседании Кабинета Министров, посвященном итогам социально-экономического развития страны в 2014 году и важнейшим приоритетным направлениям экономичсекой программы на 2015 год, говорит о том, что добиться структурных преобразований экономики невозможно без создания развитой инфраструктуры, в первую очередь, информационно-коммуникационных систем, дорожно-транспортного и инженерно-коммуникационного строительства.

Особое внимание должно быть обращено на реализацию Комплексной программы развития Национальной информационно-коммуникационной системы Республики Узбекистан на период 2013-2020 годы. Следует продолжить работы по дальнейшему увеличению технических возможностей доступа в сеть интернет, расширению оптических сетей широкополосного доступа и строительству оптоволоконных линий связи, завершить перевод всех регионов, включая труднодоступные районы, на цифровое телевидение [1]. Поэтому огромное внимание уделяется развитию телекоммуникационных технологий и цифровому телефидению.

В настоящее время развитие систем компрессии видеоданных ведется по 2 направлениям: за счет улучшения существующих методов обработки изображений. Увеличение эффективности традиционных методов сопряжено с рядом трудностей, поскольку сложные алгоритмы трансформации изображений создают дополнительный массив служебной информации, который добавляется сжатым видеоданным и при больших коэффициентах сжатия начинают превалировать над ними, тем самым снижая эффективность сжатия видеопотока. Второе направление исследований, к которому относятся нейрокомпьютеры и нейротехнологии, изучает механизмы обработки информации в нейронах головного мозга, что в перспективе даст возможность резкого уменьшения требуемого количества видеоинформации для реконструирования изображений.

Объектом исследования является видео данные ТВ изображений и их сжатие новыми методами сжатия на основе нейронных сетей.

Предметом данного исследования является анализ и подробное изучение методов обработки и сжатия видеоданных ТВ изображений с использованием перспективных нейротехнологий и нейропроцессоров, и экспериментальной оценки их эффективности.

Целью работы является аналитический анализ перспективных методов обработки изображений на основе нейропроцессоров и нейротехнологий и исследование их эффективности при обработки ТВ изображений.

Задачи диссертации заключаются в следующем:

– анализ существующих методов сжатия объемов видеоданных;

– анализ построения, принцип работы, а также аппаратно-программного обеспечения для обработки изображений с помощью нейронных сетей нейронных сетей;

– оценка эффективности полученных результатов, при использовании нейронных сетей и рекомендации по их использованию.

Методы исследования: для достижения поставленной цели в диссертационной работе использовались методы системного анализа, методы цифровой обработки сигналов, теория дискретных сигналов, теория информации, методы вейвлет-анализа и методы компьютерного моделирования, принципы работы головного мозга человека и его нервной системы, принципы построения нейронных сетей и реализация его с помощью аппаратно-программных средств.

Краткий литературный обзор по теме исследования

В настоящее время во всемирной научной литературе большое внимание уделяется сжатию информации. Основные методы и алгоритмы сжатия данных нашли свое отражение работах таких авторов как Ватолин Д., А.Ю. Тропченко, А.А. Тропченко, Сэлмон Д., Уэлстид С. [2, 11, 46], которые анализировали эти методы и их применение. Также исследованы алгоритмы архивации данных, впервые изученные аспирантом Массачусетского технологического института Дэвидом Хаффманом.

Все большее распространение альтернативные методы сжатия изображения и звука на основе вейвлет-преобразований, введенных Гроссманом и Морле в середине 1980-х годов. В настоящее время для сжатия ТВ программ широкое распространение получили стандарты вещательного ТВ MPEG-4 [10] и стандарт MPEG-4-10, принятый ISO в 2001, позволяет получить большие коэффициенты сжатия. Но на скоростях видеопотока менее 3 Мбит/с в нем проявляются искажения в виде блочного эффекта.

Изучается совершенно новые методы обработки изображений на основе нейросетевых технологий, впервые разработанные и изученные такими ученными как Розенблат, Кохоненом, Гроссбергом и Хопфилдом [18, 20, 23]. Они рассмотрели совершенно новый метод обработки информации, на основе работы головного мозга человека, а также всей его нервной системы. Изучены принципы передачи информации от мозга к нервным окончаниям и обратно, после чего была создана искусственная нейронная сеть для

обработки данных.

Научная новизна данной работы заключается в исследовании передовых технологий обработки изображений способных в перспективе обеспечить лучшие соотношения показателей качество изображений/объем данных, чем у существующих технологий.

Актуальность данного направления исследований заключается в том, что современные методы сжатия видео данных ТВ программ не могут обеспечить приемлемое визуальное качество изображений на скоростях цифрового потока менее 2,5-3 Мбит/с, особенно в форматах HD. Это связано с тем, что объем дополнительной информации, необходимый для декодирования сжатых изображений становится соизмерим с данными сжатых изображений, что снижает общую эффективность кодирования. Кроме того, сжимаемость изображений сильно зависит от структуры. Так, изображения с крупными и однородными по цвету видеообъектами сжимаются хорошо, а мелкоструктурные изображения сжимаются плохо. А поскольку часто требуется обеспечить постоянную скорость передачи данных, то для поддержания постоянного битрейта плохо сжимаемые изображения дожимаются за счет снижения их качества. Поэтому разработка новых методов большого сжатия данных ТВ программ без ухудшения качества изображений и звука имеют важное научно-практическое значение.

Теоретическая и практическая значимость результатов исследования состоят в том, что проведенные исследования позволяют выявить наиболее эффективные методы сжатия мультимедийных данных ТВ программ с наилучшим соотношением коэффициента сжатия и визуального качества восстановленных изображений, что позволит более эффективно использовать каналы связи или накопители для хранения видеофильмов.

Апробация работы. Основные положения диссертационной работы докладывались и обсуждались на следующих научно-технических конференциях**:**

- на Республиканской научно-технической конференции молодых ученых, исследователей, магистрантов и студентов на тему «Проблемы информационных технологий и телекоммуникаций», Ташкент, 14-15 марта 2013 г.;
- на Республиканской научно-технической конференции молодых ученых, исследователей, магистрантов и студентов на тему «Проблемы информационных технологий и телекоммуникаций», Ташкент, 20-22 апреля 2011 г.;
- на Республиканской научно-технической конференции молодых ученых, исследователей, магистрантов и студентов на тему «Проблемы информационных технологий и телекоммуникаций», Ташкент, 15-16 марта 2012 г.;
- на международной конференции «Актуальные проблемы развития инфокоммуникаций и информационного общества», Ташкент, 26-27 июня 2012 г.
- на Республиканской научно-технической конференции молодых ученых, исследователей, магистрантов и студентов на тему «Информационные технологии и проблемы телекоммуникаций», Ташкент, 14-15 марта 2013 г.;
- на Республиканской научно-технической конференции на тему «Перспективы эффективного развития информационных технологий и телекоммуникационных систем», Ташкент, 13-14 марта 2014 г.;
- на VII Международной научной конференции «Приоритетные направления в области науки и технологии в XXI веке», Ташкент, 30-31 мая 2014г.

Диссертационная работа состоит из введения, 4 глав, заключения и списка литературы из 44 наименований. Основной текст содержит 104 страницы и иллюстрируется 37 рисунками.

Содержание работы.

Во введении обоснована актуальность темы, научная новизна и практическая ценность исследований, изложены цель и задачи исследования выполненной работы.

В первой главе приведены основные понятия мультимедийного контента ТВ программ. Проведена классификация избыточности мультимедийной информации и проанализированы основные методы сжатия видеоинформации.

Во второй главе приведены основные понятия о нейронных сетях. Проведен анализ возможностей нейросетевых технологий, методы построения и принцип работы, а также рассмотрены существующие методы обработки изображений.

В третьей главе была исследована аппаратно-программная часть реализации искусственных нейронных сетей для обработки изображений. Проведен анализ существующих нейропроцессоров и нейроиммитаторов для обработки объемов видеоданных

В четвертой главе приведены экспериментальные данные для сжатия объемов видеоданных и их сравнение с выявлением всех их достоинств и недостатков. На основе полученных экспериментальных данных предложены рекомендации по использованию искусственных нейронных сетей в цифровом телевидении.

В заключении сформулированы теоретические и практические выводы к диссертационной работе.

Все результаты получены автором лично.

1. АНАЛИТИЧЕСКИЙ ОБЗОР СУЩЕСТВУЮЩИХ И ПЕРСПЕКТИВНЫХ МЕТОДОВ СЖАТИЯ ВИДЕОДАННЫХ ТВ ПРОГРАММ

1.1. Общие положения

С развитием телевизионного вещания возрастают потребности людей, как к качеству наблюдаемых изображений, так и к расширению возможностей телевизионных систем. Так все больше ТВ программ сейчас передается в форматах высокой четкости, что приводит к значительному увеличению объема видеоданных. Также развиваются системы интерактивного и трехмерного телевидения использующие гораздо большие скорости передачи данных. Однако цифровое телевидение использует стандартные 8 мегагерцовые каналы связи, что требует разработки более эффективных методов сжатия объемов видеоданных при сохранении визуального качества отображаемых изображений. Еще более сложная задача стоит при передаче ТВ программ по относительно узкополосным каналам Интернета и мобильной связи, где требуются коэффициенты сжатия видеопотока в 130 и более раз.

Действенным методом снижения технических требований к пропускной способности используемых каналов передачи данных, определяемой вероятностно-энергетическими характеристиками и скоростью передачи, является кодирование (сжатие) видеоданных при их передаче и декодирование в приемном устройстве. При этом становятся возможными многие виды услуг, которые требуют передачи видеосигналов в реальном времени, например, такие как видеотелефония, мобильная и стационарная телеконференцсвязь, многопрограммное интерактивное телевидение, телевидение высокой четкости, многопрограммное звуковое вещание и др. Алгоритмы сжатия изображений и видеоданных постоянно совершенствуются и создаются новые стандарты [2, 3].

Разработка алгоритмов цифрового сжатия различных видов информации для их передачи по каналам связи как альтернативы аналоговым системам проводится уже более 20 лет во всем в мире. Был получен ряд важных результатов в плане разработки алгоритмов сжатия, включая стандарты JPEG (JPEG-2000), MPEG-1, MPEG-2, MPEG-4 (видео), H.261, H.263, H.264 (AVC) и др. для статических и динамических изображений различного разрешения.

К настоящему времени исследованы разные алгоритмы сжатия изображений. Самыми популярными в практике алгоритмами сжатия изображений являются алгоритмы поблочного кодирования с преобразованием, основанные на дискретных ортогональных преобразованиях. Среди алгоритмов с преобразованием фактическим стандартом являются алгоритмы Joint Photographic Experts Group – JPEG и JPEG2000 для статических изображений и опорных кадров, а также H.264.и H.265 для видеопотоков. Однако, алгоритмы используемые в этих стандартах при больших коэффициентах сжатия порождают значительные искажения изображений, что снижает их визуальное качество [3].

В последние годы в связи с широким распространением нейросетевых методов были разработаны нейросетевые алгоритмы сжатия изображений. Исследовались разные типы нейронных сетей как обучаемые с учителем, так и самообучаемые нейронные сети. Нейронные сети могут быть легко распараллелены, что позволяет сократить время вычисления и дает возможность сжимать изображения в реальном времени.

1.2. Классификация типов избыточной информации ТВ изображений и методы ее устранения

Характерной особенностью большинства типов данных является их избыточность. Степень избыточности данных зависит от типа данных.

Например, предсказуемость значений пикселей в графике, состоящей из графических примитивов значительно выше, чем в случайное изменение яркостей реальных изображений. Поэтому графические изображения сжимаются значительно лучше (например, фрактальными методами), чем реальные изображения. Причем, следует учитывать, что избыточность изображений очень сильно зависит от их сюжета. В относительно однородных изображениях с большими видео объектами избыточности много, а в мелкоструктурных – мало, поэтому такие изображения сжимаются плохо. Другим фактором, влияющим на степень избыточности является принятая система кодирования. Примером систем кодирования могут быть обычные языки общения, которые являются ни чем другим, как системами кодирования понятий и идей для высказывания мыслей. Так, установлено, что кодирование текстовых данных с помощью средств русского языка дает в среднем избыточность на 20-25% большую, чем кодирование аналогичных данных средствами английского языка [4].

Из анализа ТВ изображений известно, что они обладают большим объемом различной избыточной информации, которую можно разделить на следующие классы [5]:

- **кодовую,**
- **межэлементную или статистическую,**
- **психовизуальную,**
- **структурную,**
- **временную или межкадровую.**

При этом сжатие информации производится вследствие устранения одного или нескольких указанных типов избыточности, которые рассмотрим подробнее.

Причиной возникновения **кодовой избыточности** является то, что изображения, как правило, состоят из объектов, имеющих регулярную форму и отражательные свойства поверхности. В результате чего, на большинстве

изображений определенные значения яркости оказываются более вероятными, чем другие. А поскольку при двоичном кодировании яркостей пикселей используются коды одинаковой длины, то появляется кодовая избыточность. Для устранения кодовой избыточности применяют энтропийное сжатие кодами переменной длины (коды Хаффмана), где часто повторяющиеся кодовые комбинации заменяются короткими кодами, а редко встречающиеся - длинными. Такой подход принято называть сжатием без потерь (энтропийным сжатием) и позволяет на 20-25% снизить объем передаваемой информации.

Причиной возникновения **межэлементной избыточности** изображений является высокая разрешающая способность дискретного поля изображения, которая реализуется только вдоль контуров, а на всех гладких участках изображения она расходуется впустую, т.е. возникает межэлементная избыточность, которая резко увеличивает объем информации.

Психофизическая избыточность основывается на особенностях нашего зрительного восприятия, которое заключается в том, что часть информации в изображении может быть исключена без заметного визуального ухудшения их качества. Так, глаз меньше замечает изменения цветности, чем яркости. Кроме того, экспериментально установлено, что при наблюдении человек стремится, в первую очередь отыскать в изображении такие его наиболее важные отличительные характеристики, как контуры или текстурные области, и образовать из них комбинации, поддающиеся распознаванию. А цвет и яркость элементов играют, по всей видимости, вспомогательную роль. При этом успех восприятия определяется организацией экономного описания таких элементов, как контур или область.

Структурная избыточность основывается на присутствии однородных участков в изображении. Для устранения контурно- текстурной избыточности используют сканирование изображения для нахождения повторяющихся его фрагментов, которые заменяются ссылками на уже

найденный фрагмент, что существенно снижает объем передаваемой информации.

Временная или межкадровая избыточность ТВ изображений определяется структурой видеопотока, поскольку информация в смежных кадрах одного видеосюжета, как правило, мало отличается. Поэтому, применяя устранение межкадровой корреляции, можно обеспечить довольно большие коэффициенты сжатия видеопотока, что реализуется в стандартах сжатия семейства MPEG и многих других кодеках. При этом структура видеопотока состоит из опорного кадра, где устраняется только внутрикадровая избыточность и одного или нескольких типов кадров передающих межкадровые различия.

На сегодняшний день разработано много различных методов и алгоритмов сжатия видеоинформации, обладающие разной эффективностью сжатия, качественными показателями, сложностью реализуемых алгоритмов и быстродействием. При этом в механизмах обработки опорного кадра можно выделить следующие направления:

- Сжатие на основе преобразования (ДКП, вейвлет, и др);
- Фрактальное сжатие;
- Векторное квантование;
- Межкадровая избыточность

У каждого подхода есть свои достоинства и недостатки, поэтому рассмотрим некоторые из них более подробно.

1.3. Методы сжатия видеоданных на основе спектральных преобразований

Одним из возможных и наиболее распространенных способов обработки, сжатия изображений и видеопоследовательностей является

применение ортогональных преобразований, в основе которых могут быть положены различные принципы [13]. Наиболее часто используются методы линейных ортогональных преобразований.

Наиболее подходящими для этой цели оказались следующие преобразования [13]:

- **преобразование Уолша-Адамара;**
- **преобразование Карунена-Лоэва;**
- **дискретное косинус-преобразование (ДКП);**
- **вейвлет-преобразования (ВП)**

У каждого из приведенных преобразований есть свои достоинства, недостатки и области применения. Так достоинством **преобразования Адамара** является простота реализации и низкая вычислительная сложность. Данное разложение дает хорошие результаты для кусочно-постоянных функций, выделяющих постоянные составляющие сигналов, однако, в реальных изображениях такие сигналы встречаются достаточно редко. Основным недостатком **преобразования Карунена-Лоэва** является то, что пока не разработан быстрый метод вычисления его векторов, поэтому данный метод пока является сугубо теоретическим.

Таким образом, из перечисленных выше преобразований на практике активно используются только ДКП и ВП, которые рассмотрим более подробно.

Дискретно-косинусное преобразование (ДКП)

ДКП является хорошо изученным и весьма эффективным преобразованием, предложенным В.Ченом в 1981 году и использующееся в форматах JPEG, MJPEG, MPEG-1, MPEG-2, MPEG-4. По сути, этот метод сходен с двумерным дискретным преобразованием Фурье и отличается от него только используемыми базисными функциями. Достоинством ДКП является быстрая сходимость ряда, что обеспечивает меньшую погрешность

ошибки преобразования [6].

Прямое и обратное ДКП описываются уравнениями (1.1 и 1.2)

$$F(u,v) = (1/4)C(u)C(v)\sum_{x=0}^{7}\sum_{y=0}^{7} p(x,y)\left[\cos\frac{(2x+1)u\pi}{16}\right]\left[\cos\frac{(2y+1)v\pi}{16}\right] \quad (1.1)$$

$$f(x,y) = \frac{2}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} C(u)C(v)F(u,v)\cos\frac{(2x+1)u\pi}{2N}\cos\frac{(2y+1)v\pi}{2N} \quad (1.2)$$

где v - горизонтальная координата графического блока, u - вертикальная, x - вертикальная координата внутри блока, а y - горизонтальная координата внутри блока, $C(u)$, $C(v) = 1/\sqrt{2}$ для $u,v = 0$ и $C(u)$, $C(v)$ **= 1** в противном случае.

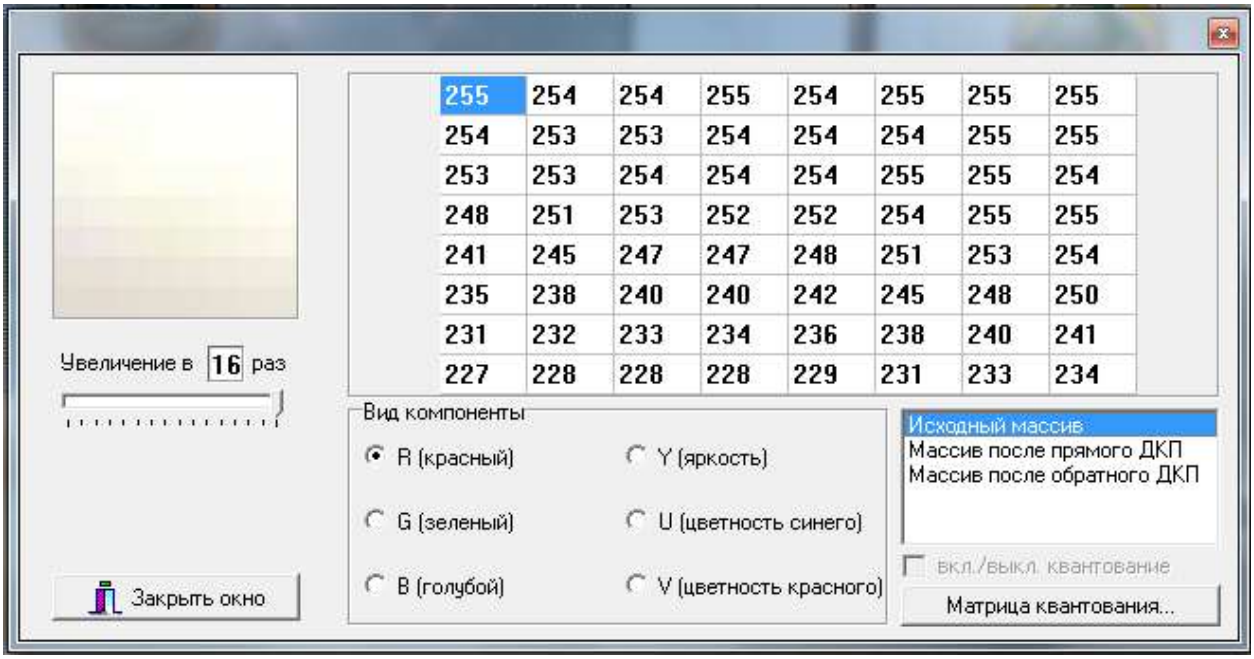
$$A(u) = \begin{cases} \frac{1}{\sqrt{2}}, & for\ u \equiv 0 \\ 1, & for\ u \neq 0 \end{cases} \quad (1.3)$$

Данный метод предусматривает разбиение кадра (рис.1.1) на блоки по **64 (8x8)** отсчета, называемые **сигнальными матрицами (**рис.1.2,а**).** После чего сигнальные матрицы преобразуются в матрицы частотных коэффициентов (рис.1.2,б**)** такого же размера, которые можно рассматривать как двумерный спектр изображения в горизонтальном и вертикальном направлениях. Причем, в такой матрице коэффициенты в левом верхнем углу соответствуют низкочастотной составляющей изображения, а в правом нижнем — высокочастотной.

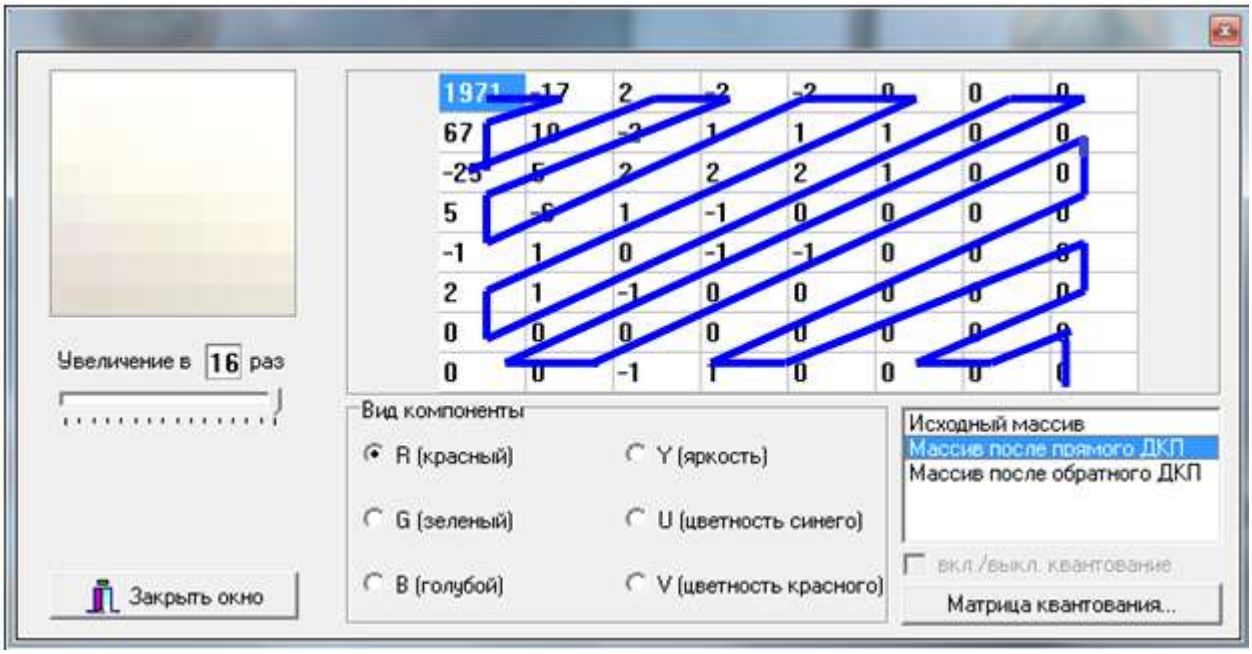
Особенность спектра ДКП состоит в том, что основная энергия частотных составляющих этого спектра концентрируется в небольшой области около нулевых частот. Амплитуда высокочастотных составляющих или мала, или просто равна нулю, поэтому передаче подлежат только те частотные коэффициенты матрицы ДКП, величины которых превышают принятые пороговые значения. Коэффициенты ниже порогового значения считаются нулевыми и не передаются для чего производится их зигзагообразное считывание (рис.1.2,б) и сжатие статистическим компрессором длинных серий (RLE).

Если при передаче отбрасываются только нулевые коэффициенты, то получается сжатие без потерь качества, т.е. после декомпрессии изображение не будет отличаться от оригинала**.** Однако при этом коэффициент сжатия не высок и в среднем составляет 10-20 раз в зависимости от детальности изображения. Для управления коэффициентом сжатия применяют деление коэффициентов ДКП на определенные числа (матрицу квантования) с последующим округлением до целого числа, что увеличивает длину цепочек нулевых коэффициентов и соответственно коэффициент сжатия. Однако, это округление данных с одной стороны приводит к увеличению сжатия изображения, с другой стороны, к необратимым потерям информации в результате чего при больших коэффициентах сжатия нарушается плавность изменения яркости на границах блоков, что приводит к возникновению искажений в виде блочного эффекта снижающего разборчивость и качество восстановленного изображения, как показано на рис.1.3.

Рис.1.1. Исходное изображение



а)



б)

Рис.1.2. Сигнальная матрица яркостей пикселей исходного изображения размером 8x8 пикселей (а) и матрица коэффициентов после прямого ДКП (б)

Данный метод обладает хорошей производительностью, хорошо сочетается с блочным методом компенсации движения и обеспечивает хорошее качество изображений при скоростях видеопотока более 5 Мбит/с. Однако, на меньших скоростях сильно проявляются искажения в виде

блочного эффекта, в результате которого изображение приобретает мозаичный вид, что является основным недостатком этого метода сжатия [10].

ДКП используется в стандартах сжатия изображений JPEG и MPEG.

Рис.1.3. Исходное и восстановленное изображение при сжатии 100 раз

Сжатие изображения на основе вейвлет-преобразований

В настоящее время получают широкое распространение альтернативные методы сжатия изображения и звука на основе Вейвлет-преобразований, введенных Гроссманом и Морле в середине 80-х годов [8]. Введение нового метода стало необходимым из-за серьезных недостатков преобразований Фурье и ДКП, которые заключались в том, что их гармонические базисные функции плохо работают с непериодическими сигналами. В результате, безвозвратно терялась некоторая часть полезной информации.

На данный момент существует множество вейвлет функций, и наиболее распространенные представлены на рис. 1.4.

Рис.1.4. Некоторые наиболее распространенные вейвлеты.

Выбор оптимального базиса вейвлетов для кодирования изображения является трудной задачей. Известен ряд критериев построения вейвлетов, среди которых наиболее важными являются: гладкость, точность аппроксимации, величина области определения, частотная избирательность фильтра [15].

Простейшим видом вейвлет-базиса для изображений является разделимый базис, получаемый сжатием и растяжением одномерных вейвлетов. Использование разделимого преобразования сводит проблему поиска эффективного базиса к одномерному случаю. Однако неразделимые базисы могут быть более эффективными.

Прототипами базисных функций для разделимого преобразования являются функции $\varphi(x)(y)$, $\varphi(x)(y)$, $(x)\varphi(y)$ и $(x)(y)$. На каждом шаге преобразования выполняется два разбиения по частоте. Предположим, что изображение имеет размерность N×N. Сначала каждая из N строк изображения делится на низкочастотную и высокочастотную половины. Получается два изображения размерами N×N/2 . Далее, каждый столбец делится аналогичным образом. В результате получается четыре изображения размерами N/2×N/2: низкочастотное по горизонтали и вертикали, высокочастотное по горизонтали и вертикали, низкочастотное по горизонтали и высокочастотное по вертикали и высокочастотное по горизонтали и низкочастотное по вертикали. Первое из вышеназванных

изображений делится аналогичным образом на следующем шаге преобразования, как показано на рис.1.4 [15].

При этом изображения, как правило, обрабатываются целиком, что уменьшает их четкость и устраняет блочный эффект (Рис 1.5) [9].

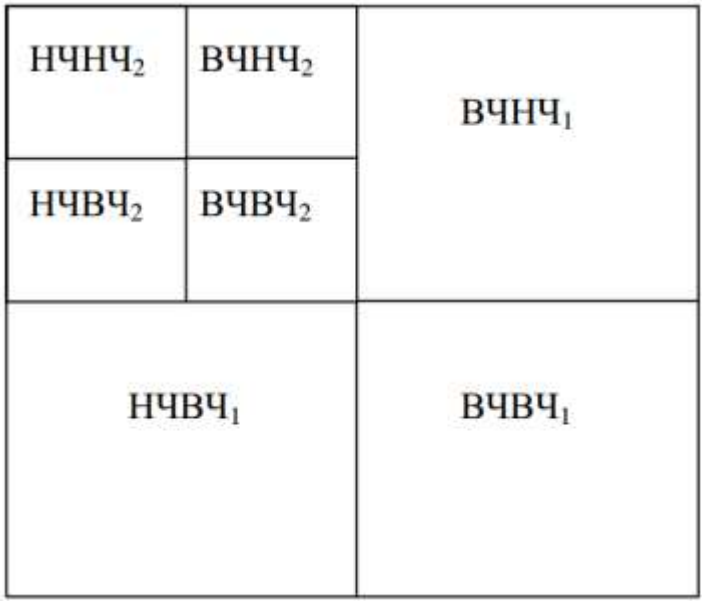


Рис.1.4. Два уровня вейвлет-преобразования изображения

А) Б) В)

Рис. 1.5. а) оригинальное изображение, б) результат декомпозиции первого уровня, в) результат декомпозиции второго уровня

Поэтому алгоритмы сжатия типа Wavelet способны обеспечивать более высокие по сравнению с алгоритмом JPEG коэффициенты сжатия, как показано на рис.1.6.

(а) (б)

Рис. 1.6. Сравнительное качество восстановленных изображений после вейвлет преобразований (а) и ДКП (б) при Ксж=100.

Как видно из рис 1.6, вейвлет-сжатие обеспечивает лучшее качество изображения при одинаковом коэффициенте сжатия. Таким образом, преимущество метода вейвлет сжатия перед ДКП состоит в том, что вейвлет преобразует полное изображение, а не его отдельные фрагменты, и позволяет получить качественное изображение при больших (до 100) коэффициентах сжатия. Однако при высокой степени компрессии и вейвлет сжатие может давать искажения, имеющие вид ряби вблизи резких границ, но такие искажения в среднем меньше бросаются в глаза наблюдателю, чем блочная «мозаика», создаваемая ДКП [7].

Вейвлет – преобразования наиболее часто применяются для обработки изображений высокого пространственного разрешения, характеризующегося большой степенью корреляции между соседними пикселями [9].

Поскольку в процессе вейвлет-преобразований изображение не делится на блоки как показано на рис 1.5, то к нему сложно применить методы компенсации движения, обеспечивающие основное сжатие в стандартах MPEG. Поэтому, вейвлет-преобразование в основном применяются для сжатия статических изображений в стандарте JPEG-2000 и MPEG-4.

1.4. Методы сжатия видеоданных на основе фрактального кодирования

Для получения высоких коэффициентов сжатия порядка 200-2000 могут использоваться фрактальные методы сжатия изображений. Основой метода является рассмотрение естественных объектов как «подобных самим себе» и подчиняющихся требованиям фрактальной геометрии, в которой сложные структуры выглядят точно так же, как и простые структуры, т.е. повторяют их. Задачей кодирования является отыскание таких совпадений в цифровых изображениях и описание таких фракталов с дальнейшим эффективным повторением.

Понятия фракталов были предложены математиком ***Б. Мандельбротом*** в 1975 г. для обозначения нерегулярных, но самоподобных структур, для которых некоторые свойства реального изображения, сохраняются при масштабировании пространства (Рис 1.7). При фрактальном кодировании используется свойство подобия деталей разного масштаба, встречающиеся в реальных изображениях [11].

Фрактальная архивация основана на том, что изображение представляется в более компактной форме с помощью коэффициентов системы итерационных функций (IFS) (итерация - повторное применение математической операции в серии аналогичных операций, производимых для получения результата). Система итерирующих функций – это совокупность сжимающих аффинных преобразований, которые включают в себя масштабирование, поворот и параллельный перенос.

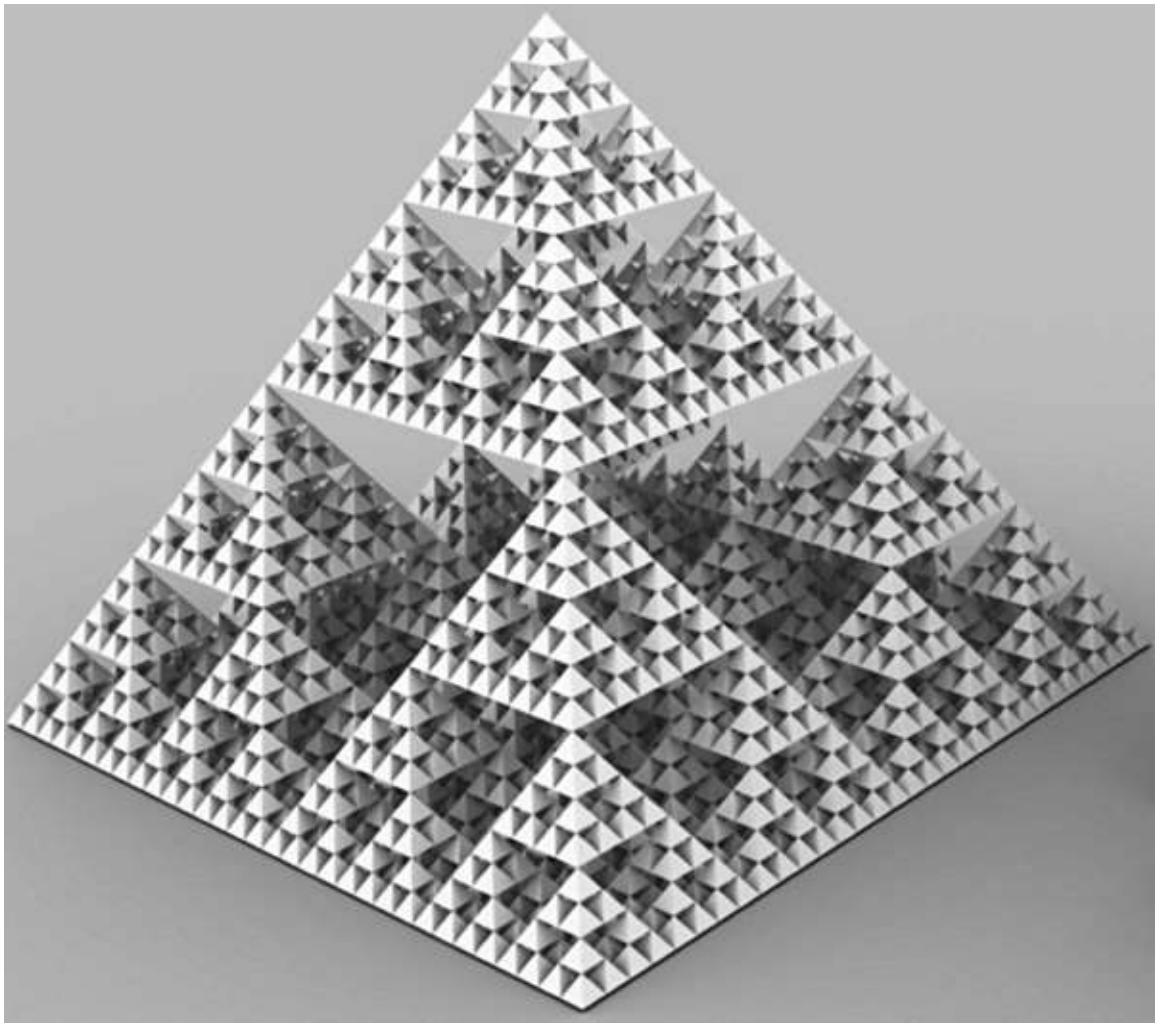


Рис 1.7. Пример использования фракталов для построения пирамиды.

Из курса линейной алгебры известна формула вычисления новых координат X',Y' при аффинных преобразованиях [11]:

$$X` = x * a - y * b + e$$

$$Y` = x * c + y * d + f$$

Здесь

a = cos (alpha) * scale_x,

b = sin (alpha) * scale_x,

c = sin (alpha) * scale_y,

d = cos (alpha) * scale_y,

e = move_x (смещение),

f = move_y,

где

scale_x - масштабирование по оси X;
scale_y - масштабирование по оси Y;
alpha - угол поворота;
move_x - параллельный перенос по оси X;
move_y - параллельный перенос по оси Y.

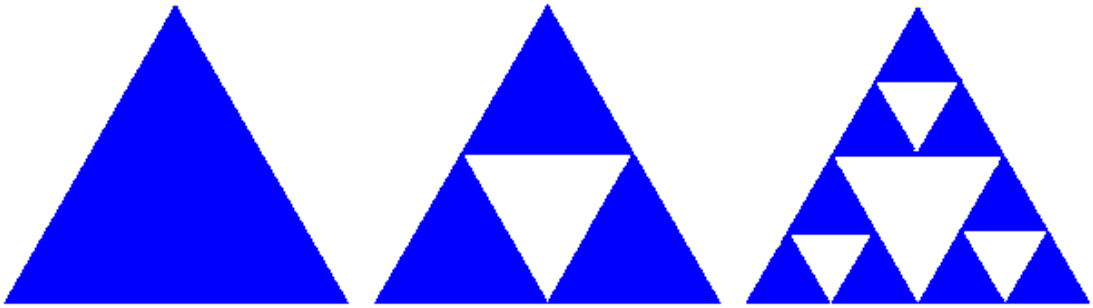
Полученные коэффициенты a, b, c, d, e, f для каждого элемента разбиения и составят требуемую систему итерирующих функций.

Афинное преобразование считается сжимающим, если коэффициент масштабирования меньше единицы, например y=0.5x.

Сжимающие преобразования обладают важным свойством. Если взять любую точку и начать итеративно применять к ней одно и то же сжимающее преобразование: f(f(f...f(x))), то результатом будет всегда одна и та же точка. Чем больше раз применяется преобразование, тем точнее находится эта точка, которая называется ***неподвижной точкой.*** Несколько аффинных сжимающих преобразований образуют систему **итерированных функций (СИФ),** которые по сути представляют множительный механизм многократно искажающий и перемещающий исходное изображение. Например, при помощи СИФ из трех функций (рис.1.8) можно построить треугольник Серпинского [11] .

Как видно из рисунка 1.9, исходный треугольник три раза множится, уменьшается и переносится. И так далее. Если так продолжать до бесконечности, получится фрактал Серпинского с площадью 0 и размерностью 1,585. Причем вместо треугольника можно использовать квадрат, круг или другие геометрические примитивы. При этом, для кодирования объекта изображения достаточно сохранить функцию данного фрактала с указанием: в каких координатах, параметрами масштабирования и

ориентации нужно синтезировать данный объект. Таким образом, довольно крупный объект можно представить в виде 10-20 байтов его параметров и тем самым существенно снизить объем передаваемой информации.



$$f_1\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$f_2\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1/2 \\ 0 \end{bmatrix}$$

$$f_3\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1/4 \\ \sqrt{3}/4 \end{bmatrix}$$

Рис.1.8. Применение СИФ для построения треугольника Серпинского

Фрактальное сжатие хорошо работает на изображениях технических чертежей, текста, карт местности и изображениях компьютерной графики (рис.1.11). При этом коэффициент сжатия таких изображений может достигать **200-2000 раз** [11].

На реальных ТВ изображениях фрактальное сжатие тоже дает хорошие результаты, однако, на поиск фракталов с учетом масштабирования и взаимной ориентации требуются очень большие затраты времени (иногда до 5-10 минут на кадр, что для часовой передачи из 90 000 кадров потребует более **7,5 тысяч часов** [47]**.** Поэтому для телевизионных программ фрактальные методы сжатия пока не применяются.

Рис.1.9. Применение фракталов для синтеза сложных изображений

1.5. Методы сжатия видеоданных на основе компенсации движения

В настоящее время существует большое разнообразие методов сжатия видеоданных, отличающихся коэффициентом сжатия видеопотока, качеством восстановленных изображений, быстродействием обработки изображений, причем, для телевидения необходимо обеспечить обработку изображений в реальном масштабе времени. Поэтому проблемы эффективного сжатия сигналов ТВ изображений имеют важное практическое значение.

Следует отметить, что поскольку в пределах одного видеосюжета межкадровые различия смежных кадров, как правило, малы (рис.1.10), то в ТВ стандартах сжатия MPEG и других видео кодеках основное сжатие видеопотока обеспечивается за счет устранения межкадровой избыточности [16].

Рис.1.10. Изображения 2-х смежных кадров и их межкадровая разность

В настоящее время в стандартах MPEG межкадровая избыточность устраняется при помощи методов компенсации движения (КД), которые компенсируют перемещение прямоугольных областей текущего кадра. При этом для каждого фиксированного блока, состоящего из M*N пикселей обрабатываемого кадра, выполняется следующая процедура [16]:

1. Поиск на ссылочном кадре (предыдущем или следующем), ранее закодированном и переданном декодеру, «подходящего» блока из M*N пикселей. Это делается путем сравнения фиксированного M*N-блока с некоторыми или со всеми блоками M*N области поиска. Область

поиска обычно представляет собой некий регион с центром в середине этого выбранного блока. Популярным критерием схожести блоков является энергия остатка, получаемая вычитанием блока-кандидата из фиксированного M*N блока, то есть выбирается блок-кандидат, минимизирующий энергию остатка. Этот процесс поиска подходящего блока называется оценкой движения.

2. Выбранный кандидат становится прогнозом текущего M*N-блока, и его необходимо вычесть из этого блока для получения остаточного M*N-блока.

3. Остаточный блок кодируется и передается декодеру, и декодер получает координаты вектора смещения текущего блока по отношению к позиции блока-кандидата – вектора движения.

Декодер использует вектор движения для нахождения блока-прогноза, декодирует остаточный блок, и складывает его с прогнозом для реконструкции версии исходного блока.

Таким образом, например, 256 байтов блока размером 16x16 пикселей можно заменить 1-2 байтами его новых координат. При этом структура видеопотока состоит из опорного кадра, где устраняется только внутрикадровая избыточность и одного или нескольких типов кадров, передающих межкадровые различия и векторы перемещений блоков [16]. На рис.1.11 показаны изображения смежных кадров и результат компенсации движения их объектов, где передачи подлежат только цветные блоки, а черные заменяются на значения их новых координат.

Рис.1.11. Компенсация движения видео объектов на основе перемещения блоков.

На сегодняшний день, существует довольно много различных методов КД, обладающих различной эффективностью и быстродействием [16]. Наиболее простые используют разделение изображения на прямоугольные блоки фиксированного размера (MPEG-1, MPEG-2), однако, такая конфигурация блоков плохо согласуется с криволинейными очертаниями объектов изображения, что снижает точность компенсации и соответственно их эффективность.

Несмотря на внешнюю простоту принципа оценки и компенсации движения, его практическая реализация сталкивается со значительными трудностями, главная из которых – это многообразие сюжетов, не подвергающихся формализации.

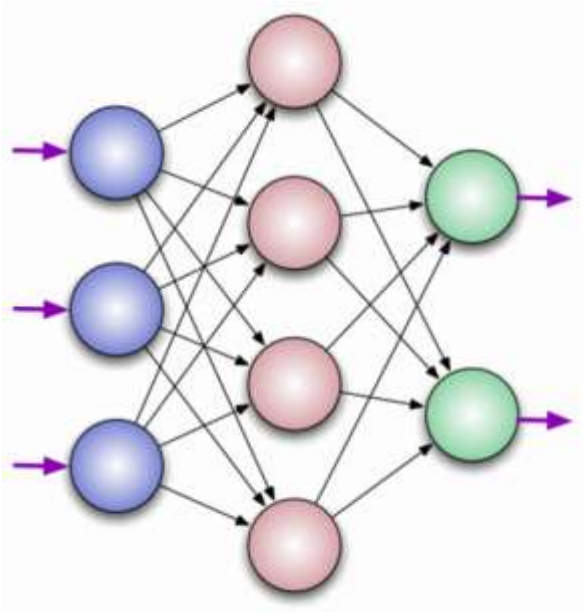
1.5. Нейросетевые методы обработки изображений

В настоящее время у ученых большинства развитых стран мира развивается неподдельный интерес к принципам функционирования человеческого организма, в частности, головного мозга и нервной системы человека, а также применение полученных результатов в различных сферах науки и техники [12].

Известно, что нервная система и мозг человека состоят из нейронов, соединенных между собой нервными волокнами, которые способны передавать электрические импульсы между нейронами. Таким образом, человек получает информацию обо всех процессах, которые происходят в его организме.

Основываясь на работе человеческого организма, были созданы искусственные нейронные сети (Рис 1.12), которые на данный момент широко применяются в медицине, метеорологии и других сферах деятельности, в частности, особое внимание в моей работе уделяется в исследовании применения нейронных сетей в области телевидения [12].

а)



б)

Рис 1.12. Пример изображения а) биологической нейронной сети;
б) исскуственной нейронной сети

Действительно, **актуальность применения нейронных сетей многократно возрастает тогда, когда появляется необходимость решения *плохо формализованных* задач [13].**

В настоящее время разработан ряд нейронных сетей, отличаются по своей архитектуре и алгоритму обучения и областям применения. При этом алгоритм сжатия данных состоит из трех основных аспектов [14]:

1. Построение сети, которая предусматривает общее количество нейронов в нейронной сети, количество входных и выходных нейронов, а также количество скрытых слоев.

2. Обучение сети, то есть – корректировка весов связей, в результате которой каждое входное воздействие приводит к формированию соответствующего выходного сигнала. Цель обучения состоит в выборе такого вектора весов *w*, чтобы минимизировать ошибку между фактическими реакциями нейрона *yi* и ожидаемыми значениями *zi* . Поэтому для обучения нейронной сети используется информация о текущем и ожидаемом значении выходного сигнала. Минимизация различий между фактическими реакциями

нейрона *yi* и ожидаемыми значениями *zi* может быть представлена как минимизация конкретной функции погрешности (целевой функции) $E(w)$, чаще всего определяемой как [14]:

$$E(w) = \sum_{k=1}^{p} (y_i^{(k)}(w) - z_i^{(k)})^2 \quad (1.4)$$

где p — количество предъявляемых обучающих выборок.

Целевая функция $E(w)$ в общем случае является нелинейной функцией. По причине ее сложности часто используют итерационные алгоритмы для эффективного поиска решения в пространстве состояний. При этом алгоритм стартует от исходного значения w, которое потом корректируется.

Последующая итерация w, обозначенная как wn , определяется, как очередной шаг от текущей точки wc в направлении вектора d [14],

$$w_n = w_c + \eta d \quad (1.5)$$

где положительное число η является величиной шага, показывающего, с какой скоростью мы движемся в направлении d. Каждый алгоритм обучения имеет свою схему корректировки весов нейронной сети.

3. Сжатие информации. На этом этапе в нейронную сеть подается информация, которую необходимо обработать и на выходе мы получаем готовый результат.

Установлено, что основным недостатком нейронной сети является то, что в некоторых случаях необходимо потратить некоторое количество времени для обучения сети, но этот недостаток легко решается путем использования уже имеющихся нейронных сетей.

Основными интересными на практике возможностями нейронных сетей являются следующие [33]:

- Гибкость структуры: можно различными способами комбинировать элементы нейросети (нейроны и связи между ними). За счёт этого на одной "элементной базе" и даже внутри "тела" одного нейрокомпьютера можно создавать совершенно разные вычислительные схемы, подбирать оптимальное для конкретной задачи число нейронов и слоёв сети.

- Быстрые алгоритмы обучения нейронных сетей: нейросеть даже при сотнях входных сигналов и десятках-сотнях тысяч эталонных ситуаций может быть почти мгновенно обучена на обычном компьютере. Поэтому применение нейронных сетей возможно для решения широкого круга сложных задач прогноза, классификации и диагностики.

- Возможность работы при наличии большого числа неинформативных, избыточных, шумовых входных сигналов – отсутствует необходимость в предварительном анализ данных, нейросеть сама определит их малопригодность для решения задачи и может их явно отбросить.

- Возможность работы со скоррелированными независимыми переменными, с разнотипной информацией (измеренной в непрерывнозначных, дискретнозначных, номинальных, булевых шкалах), что часто доставляет затруднение методам статистики.

- Нейронная сеть одновременно может решать несколько задач на едином наборе входных сигналов − имея несколько выходов, прогнозировать значения нескольких показателей.

- Алгоритмы обучения накладывают достаточно мало требований на структуру нейронной сети и свойства нейронов. Поэтому при наличии экспертных знаний или в случае специальных требований можно целенаправленно выбирать вид и свойства нейронов, собирать структуру нейронной сети вручную из отдельных элементов, и задавать для каждого из них нужные характеристики или ограничения.

- Нейросеть может обучиться решению задачи, которую человек-эксперт решает недостаточно точно. Обученная сеть может быть представлена в виде явного алгоритма решения задачи.

- Синтезированная (обученная) нейросеть обладает устойчивостью к отказам отдельных элементов (нейронов) и линий передачи информации в ней. Можно применять и специальные методы для повышения отказоустойчивости. Это бывает востребованным при аппаратных

реализациях сетей − для обеспечения построения надёжных систем из ненадёжных элементов.

- Высокая потенциальная параллельность вычислений (например, одновременное параллельное функционирование нейронов некоторого слоя сети) позволяет эффективно задействовать возможности современной вычислительной техники (от использования SIMD-команд до многопоточности и многопроцессорности) − что ускоряет процессы нейромоделирования и/или позволяет использовать синтезированные модели для решения задач реального времени.

Описанные возможности в основном относятся к многослойным нейронным сетям, обучаемым алгоритмом обратного распространения ошибки, и растущим (конструктивным) сетям на основе вариантов метода каскадной корреляции. Но существуют и другие типы нейронных сетей − нейросети ассоциативной памяти, нейросети для квантования данных, сжатия данных путем построения главных независимых компонент, для разделения смеси сигналов и др. Т.е. круг задач, решаемых нейросетями, очень широк, поскольку широк сам набор нейросетевых алгоритмов и технологий.

Выводы

В результате анализа существующих методов и алгоритмов сжатия изображений было установлено:

1. Существует сжатие информации с потерями и без потерь.
2. Применение сжатия без потерь видеоданных нецелесообразно, так как оно обеспечивает сжатие всего до 3-5 раз в зависимости от структуры изображения, в то время как сжатие с потерями обеспечивает компрессию в 100-200 раз и даже больше. Однако, благодаря особенностям человеческого зрительного восприятия, частичная потеря видеоданных визуально может быть незаметна.

3. Для сжатия изображений используются методы, основанные на спектральных преобразованиях, к которым относятся ДКП и вейвлет-преобразования, а также методы использующие фракталы. Основное сжатие видеопотока обеспечивается за счет передачи только межкадровых различий.
4. Спектральные преобразования приводят к существенным потерям информации в зависимости от коэффициента сжатия. В результате ДКП возникают искажения в виде блочного эффекта за счет нарушения плавности распределения яркости пикселей на границах блоков. Вейвлет-преобразования устраняют причину возникновения блочных искажений, но не позволяет эффективно использовать блочные методы компенсации движения.
5. Фрактальное сжатие хорошо работает на изображениях технических чертежей, текста, карт местности и изображениях компьютерной графики. С его помощью можно получить сжатие до 2000 раз, но оно не может работать в реальном масштабе времени.
6. В результате сжатия на основе компенсации движения за счет передачи межкадровой разницы относительно некого опорного или промежуточного кадра, объем информации существенно снижается и можно получить довольно большие коэффициенты сжатия видеопотока.
7. Нейротехнологии позволяют обеспечивать сжатие информации за счет обучения нейронной сети.

2. АНАЛИЗ ВОЗМОЖНОСТЕЙ НЕЙРОСЕТЕВЫХ ТЕХНОЛОГИЙ ДЛЯ СЖАТИЯ ОБЪЕМОВ ВИДЕО ДАННЫХ

2.1. Общие положения

Нейронные сети возникли из исследований в области искусственного интеллекта, а именно, из попыток воспроизвести способность биологических нервных систем обучаться и исправлять ошибки, моделируя низкоуровневую структуру мозга (Patterson, 1996) [17]. Основной областью исследований по искусственному интеллекту в 60-е - 80-е годы были экспертные системы. Такие системы основывались на высокоуровневом моделировании процесса мышления (в частности, на представлении, что процесс нашего мышления построен на манипуляциях с символами). Скоро стало ясно, что подобные системы, хотя и могут принести пользу в некоторых областях, не ухватывают некоторые ключевые аспекты человеческого интеллекта. Согласно одной из точек зрения, причина этого состоит в том, что они не в состоянии воспроизвести структуру мозга. Чтобы создать искусственных интеллект, необходимо построить систему с похожей архитектурой.

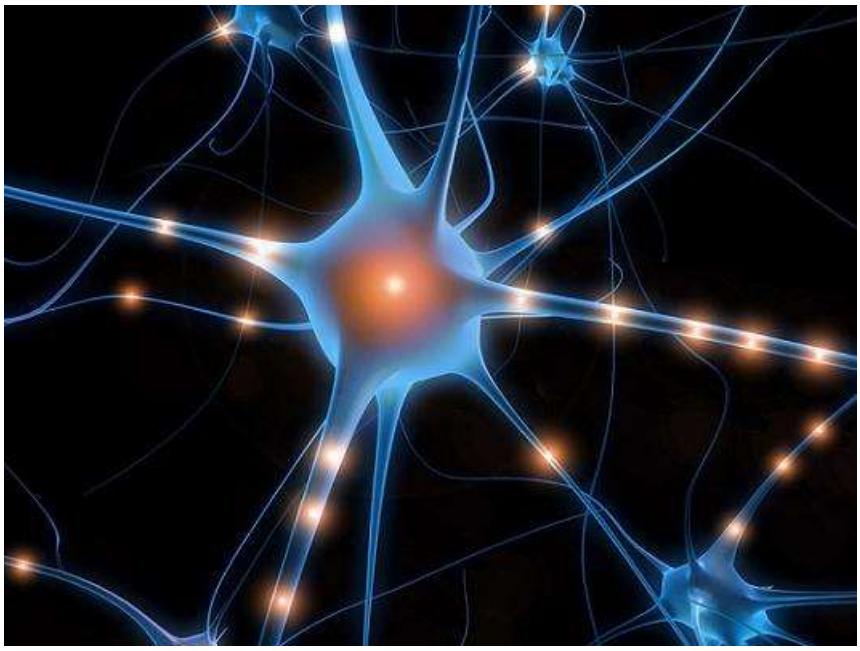


Рис 2.1. Внешний вид биологического нейрона

Мозг состоит из очень большого числа (приблизительно 10,000,000,000) *нейронов* (Рис 2.1), соединенных многочисленными связями (в среднем несколько тысяч связей на один нейрон, однако это число может сильно колебаться) [17]. Нейроны - это специальная клетки, способные распространять электрохимические сигналы (Рис.2.2). Нейрон имеет разветвленную структуру ввода информации (дендриты), ядро и разветвляющийся выход (аксон). Аксоны клетки соединяются с дендритами других клеток с помощью синапсов. При активации нейрон посылает электрохимический сигнал по своему аксону. Через синапсы этот сигнал достигает других нейронов, которые могут в свою очередь активироваться. Нейрон активируется тогда, когда суммарный уровень сигналов, пришедших в его ядро из дендритов, превысит определенный уровень (порог активации).

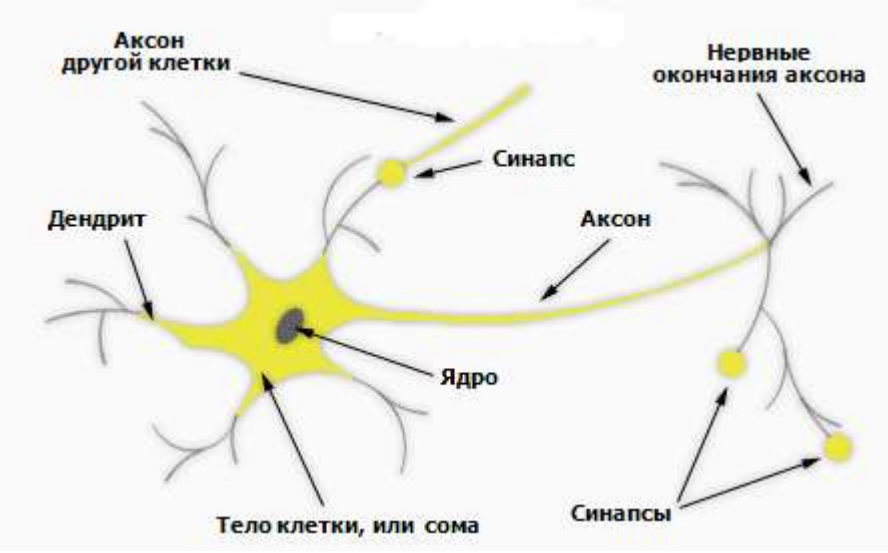


Рис 2.2. Структура биологического нейрона

Интенсивность сигнала, получаемого нейроном (а следовательно и возможность его активации), сильно зависит от активности синапсов. Каждый синапс имеет протяженность, и специальные химические вещества передают сигнал вдоль него. Один из самых авторитетных исследователей нейросистем, Дональд Хебб, высказал постулат, что обучение заключается в

первую очередь в изменениях весовых коэффициентов синаптических связей [17].

Таким образом, будучи построен из очень большого числа совсем простых элементов (каждый из которых берет взвешенную сумму входных сигналов и в случае, если суммарный вход превышает определенный уровень, передает дальше двоичный сигнал), мозг способен решать чрезвычайно сложные задачи.

Искусственный нейрон, также как и его естественный прототип, имеет группу синапсов (входов), которые соединены с выходами других нейронов, а также аксон – выходную связь данного нейрона – откуда сигнал возбуждения или торможения поступает на синапсы других нейронов [18]. Общий вид искусственного нейрона представлен на рис 2.3.

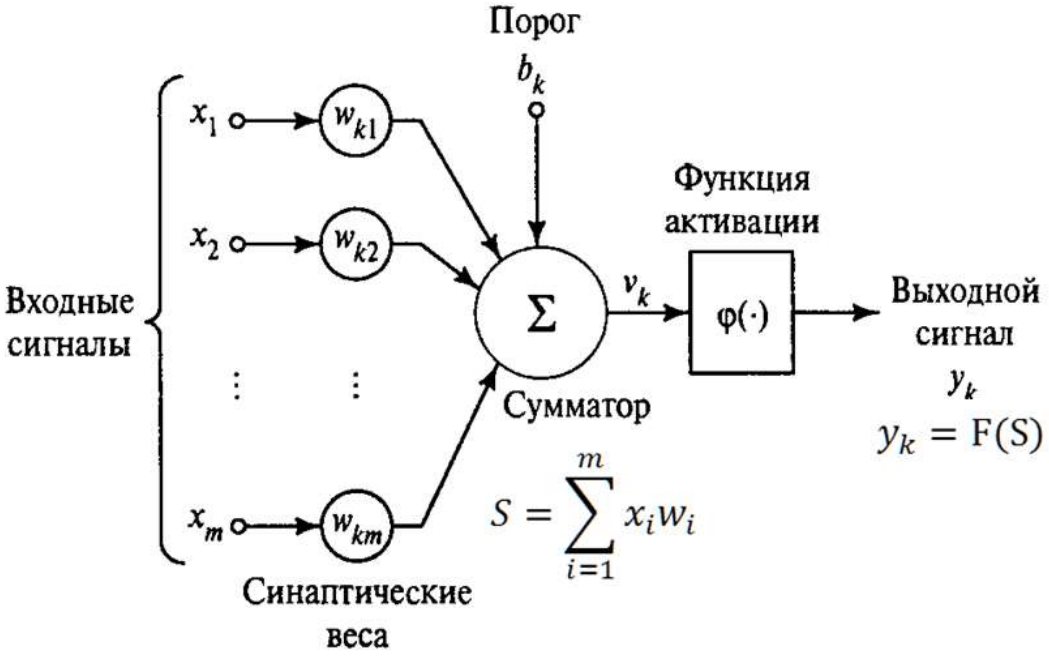


Рис. 2.3. Общий вид искусственного нейрона

Каждый синапс характеризуется величиной *синаптической связи* или *весом* w_i, который по своему физическому смыслу эквивалентен электрической проводимости.

Текущее состояние нейрона определяется как взвешенная сумма его входов [18]:

$$s = \sum_{i=1}^{n} x_i w_i \tag{2.1}$$

где x – вход нейрона, а *w* – соответствующий этому входу вес.

Выход нейрона есть функция его состояния, т.е.

$$y = f(s) \tag{2.2}$$

Нелинейная функция f(s) называется *активационной, сжимающей* функцией или функцией *возбуждения* нейрона.

Основные разновидности активационных функций, применяемых в нейронных сетях [18]., представлены на рис. 2.4.

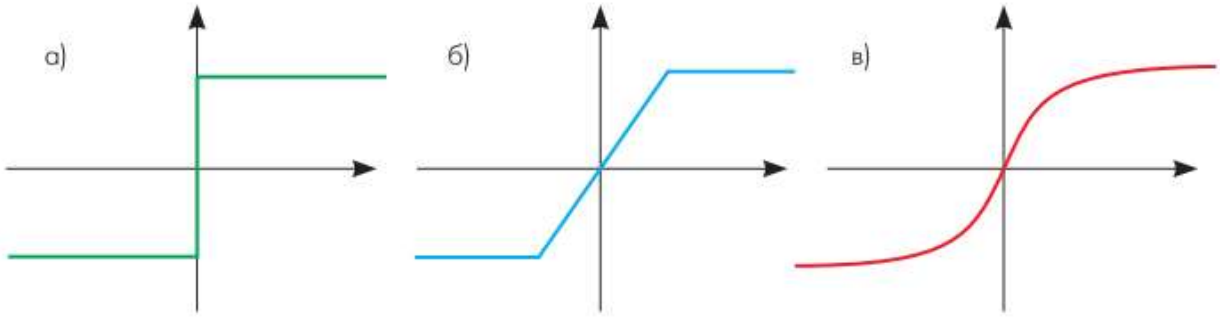


Рис 2.4. Разновидности активационных функций:
А) пороговая; Б) полулинейная; В) сигмоидальная

Пороговая функция или функция Хевисайда представляет собой перепад. До тех пор пока взвешенный сигнал на входе нейрона не достигает некоторого уровня T — сигнал на выходе равен нулю. Как только сигнал на входе нейрона превышает указанный уровень — выходной сигнал скачкообразно изменяется на единицу. Самый первый представитель слоистых искусственных нейронных сетей — перцептрон, который состоял исключительно из нейронов такого типа. Математическая запись этой функции выглядит так [19]:

$$f(x) = \begin{cases} 1 & \text{if } x \geq T \\ 0 & \text{else} \end{cases} \tag{2.3}$$

Здесь $T = -w_0 x_0$ — сдвиг функции активации относительно горизонтальной оси, соответственно под x следует понимать взвешенную сумму сигналов на входах нейрона без учёта этого слагаемого. Ввиду того, что данная функция не является дифференцируемой на всей оси абсцисс, её нельзя использовать в сетях, обучающихся по алгоритму обратного распространения ошибки и другим алгоритмам, требующим дифференцируемости передаточной функции.

Линейная функция. Сигнал на выходе нейрона линейно связан со взвешенной суммой сигналов на его входе [19].

$$f(x) = tx \tag{2.4}$$

где t — параметр функции. В искусственных нейронных сетях со слоистой структурой нейроны с передаточными функциями такого типа, как правило, составляют входной слой. Кроме простой линейной функции могут быть использованы её модификации. Например полулинейная функция (если её аргумент меньше нуля, то она равна нулю, а в остальных случаях, ведет себя как линейная) или шаговая (линейная функция с насыщением), которую можно выразить формулой [19]:

$$f(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ 1 & \text{if } x \geq 1 \\ x & \text{else} \end{cases} \tag{2.5}$$

При этом возможен сдвиг функции по обеим осям.

Недостатками шаговой и полулинейной активационных функций относительно линейной можно назвать то, что они не являются дифференцируемыми на всей числовой оси, а значит не могут быть использованы при обучении по некоторым алгоритмам.

Сигмоидальная функция. В качестве активационной функции часто используется *сигмоидальная (s-образная* или *логистическая)* функция,

показанная на рис. 3.3 (в). Эта функция математически выражается по формуле [18]:

$$f(x) = \frac{1}{1 + e^{-\alpha x}} \tag{2.6}$$

При уменьшении α сигмоидальная функция становится более пологой, в пределе при α=0 вырождаясь в горизонтальную линию на уровне 0,5; при увеличении α сигмоидальная функция приближается по внешнему виду к функции единичного скачка с порогом *T* в точке x=0. Из выражения для сигмоидальной функции видно, что выходное значение нейрона лежит в диапазоне [0,1]. Одно из полезных свойств сигмоидальной функции – простое выражение для ее производной [18]:

$$f(x) = \alpha f(x)(1 - f(x)) \tag{2.7}$$

Следует отметить, что сигмоидальная функция дифференцируема на всей оси абсцисс, что используется в некоторых алгоритмах обучения. Кроме того, сигмоидальная функция обладает свойством усиливать малые сигналы лучше, чем большие, тем самым предотвращая насыщение от больших сигналов, так как они соответствуют областям аргументов, где сигмоидальная функция имеет пологий наклон.

Выбор структуры нейронной сети осуществляется в соответствии с особенностями и сложностью задачи. Для решения некоторых отдельных типов задач уже существуют оптимальные, на сегодняшний день конфигурации. Если же задача не может быть сведена ни к одному из известных типов, то необходимо решать сложную проблему синтеза новой конфигурации.

Теоретически число слоев и число нейронов в каждом слое нейронной сети может быть произвольным, однако фактически оно ограничено ресурсами компьютера или специализированной микросхемы, на которых обычно реализуется нейронная сеть [18].

Отличительной чертой нейронных сетей является то, что место программирования в данном случае занимает обучение. Способность к обучению является фундаментальным свойством мозга. В контексте искусственных нейронных сетей процесс обучения может рассматриваться как настройка архитектуры сети, а также весов связей для эффективного выполнения поставленной задачи [20]. Обычно нейронная сеть должна настроить веса по предоставленным обучающим примерам. Свойство сети обучаться на примерах делает их более привлекательными по сравнению с системами, которые работают по заранее заложенным правилам.

Среди всех существующих методов обучения можно выделить два класса [20]:

1. Детерминированный
2. Стохастический

Детерминированный метод итеративно корректирует параметры сети, основываясь на ее текущих параметрах, величинах входов, фактических и желаемых выходов. Примером данного метода обучения является метод обратного распространения ошибки.

Стохастические методы обучения изменяют параметры сети случайным образом. При этом сохраняются только те изменения, которые привели к улучшениям. В качестве примера можно привести следующий алгоритм [20]:

1. Выбрать параметры сети случайным образом и подкорректировать их на небольшую случайную величину. Предъявить множество входов и вычислить получающиеся выходы.
2. Сравнить эти выходы с желаемыми и вычислить разницу между ними. Эта разница называется ошибкой. Цель обучения состоит в том, чтобы минимизировать ошибку.
3. Если ошибка уменьшилась коррекция сохраняется, в противном случае коррекция отбрасывается и выбирается новая.

Шаги 2 и 3 повторяются до тех пор, пока сеть не обучится.

Обучение с учителем [20]

Алгоритм называется алгоритмом обучения с учителем, если во время обучения сеть располагает правильными ответами на каждый входной пример, т.е. заранее задается множество пар векторов. В процессе обучения сеть меняет свои параметры таким образом, чтобы давать нужное отображение. Необходимо отметить, что размер множества векторов должен быть достаточным для того, чтобы алгоритм обучения смог сформировать нужное отображение [20].

Обучение без учителя

Алгоритм обучения без учителя может применяться тогда, когда известны только входные сигналы. На их основе сеть учится давать наилучшие значения выходов. Обычно алгоритм подстраивает параметры так, чтобы сеть выдавала одинаковые результаты для достаточно близких входных значений [20].

Метод Хэбба

На основании физиологических и психологических исследований Хэбб выдвинул гипотезу о том, что вес соединения между двумя нейронами усиливается, если оба эти нейрона возбуждены.

Хэбб опирался на следующие нейрофизиологические наблюдения: если связанные между собой нейроны активируются одновременно и регулярно, то сила связи возрастает. Важной особенностью этого правила является то, что изменение веса связи зависит только от активности нейронов, которые соединены данной связью.

Сам алгоритм выглядит следующим образом [20]:

1. На стадии инициации всем весовым коэффициентам присваиваются небольшие случайные значения

2. На вход сети подается входной сигнал и вычисляется выход

3. На основании полученных выходных значений нейронов производится изменение весовых коэффициентов.

4. Повтор с шага 2 с новым образом из входного множества до тех пор,

пока выходные значения сети не стабилизируются с заданной точностью.

Правило коррекции по ошибке

В 1957г. Розенблатт разработал модель, которая использует алгоритм обучения с учителем. Несмотря на некоторые ограничения, она стала основой для многих современных наиболее сложных алгоритмов обучения с учителем.

Суть алгоритма состоит в следующем [20]: для каждого входного примера задается желаемый выход. Если реальный выход сети не совпадает с желаемым, то параметры сети будут скорректированы. Для вычисления величины коррекции используется разница между реальным и желаемым выходом сети, причем коррекция весов будет происходить, только в случае ошибочного ответа.

Обучение методом соревнования

В отличие от обучения Хэбба, в котором множество выходных нейронов могут возбуждаться одновременно, при соревновательном обучении выходные нейроны соревнуются между собой за активацию. То есть из всего множества выходных нейронов используется только один нейрон с самым большим выходом. Такой алгоритм напоминает процесс обучения биологических нейронных сетей. Обучение методом соревнования позволяет классифицировать входные данные: похожие примеры группируются сетью в один класс и представляются одним образцовым элементом. При этом каждый нейрон из множества выходных нейронов «отвечает» только за один класс. Очевидно, что общее число классов, с которыми способна работать сеть равно количеству выходных нейронов. При обучении модифицируются только веса «победившего» нейрона. Это приводит к тому, что образцовый элемент становиться чуть ближе к входному примеру [20].

В настоящее время, проводятся исследования по использованию нейро сетей для сжатия распознавания образов и сжатия объемов данных. Для этой цели уже разработано несколько методов, которые рассмотрим более подробно.

2.2. Ассоциативные сети Хопфилда

В настоящее время для обработки больших объемов информации в задачах распознавания и обработке изображений начинают применятся различные искусственные нейронные сети (ИНС), под которыми понимается вид математических моделей построенных по принципу организации и функционирования их биологических аналогов – сетей нервных клеток (нейронов) мозга [21].

Модель Хопфилда (J.J.Hopfield, 1982) занимает особое место в ряду нейросетевых моделей. В ней впервые удалось установить связь между нелинейными динамическими системами и нейронными сетями. Образы памяти сети соответствуют устойчивым предельным точкам (аттракторам) динамической системы. Особенно важной оказалась возможность переноса математического аппарата теории нелинейных динамических систем (и статистической физики вообще) на нейронные сети. При этом появилась возможность теоретически оценить объем памяти сети Хопфилда, определить область параметров сети, в которой достигается наилучшее функционирование [22].

Сеть Хопфилда использует три слоя: ***входной***, ***слой Хопфилда*** и ***выходной слой*** [23]. Каждый слой имеет одинаковое количество нейронов. Входы слоя Хопфилда подсоединены к выходам соответствующих нейронов входного слоя через изменяющиеся *веса соединений*. Выходы слоя Хопфилда подсоединяются ко входам всех нейронов слоя Хопфилда, за исключением самого себя, а также к соответствующим элементам в выходном слое. В режиме функционирования, сеть направляет данные из входного слоя через фиксированные веса соединений к слою Хопфилда. Структурная схема сети Хопфилда представлена на рис. 2.5.

Сеть Хопфилда состоит из **N** искусственных нейронов. Граница **ёмкости памяти для сети** (то есть количество образов, которое она может запомнить) составляет приблизительно **15%** от числа нейронов в слое Хопфилда

(N*0,15) [23]. При этом запоминаемые образы не должны быть сильно коррелированны.

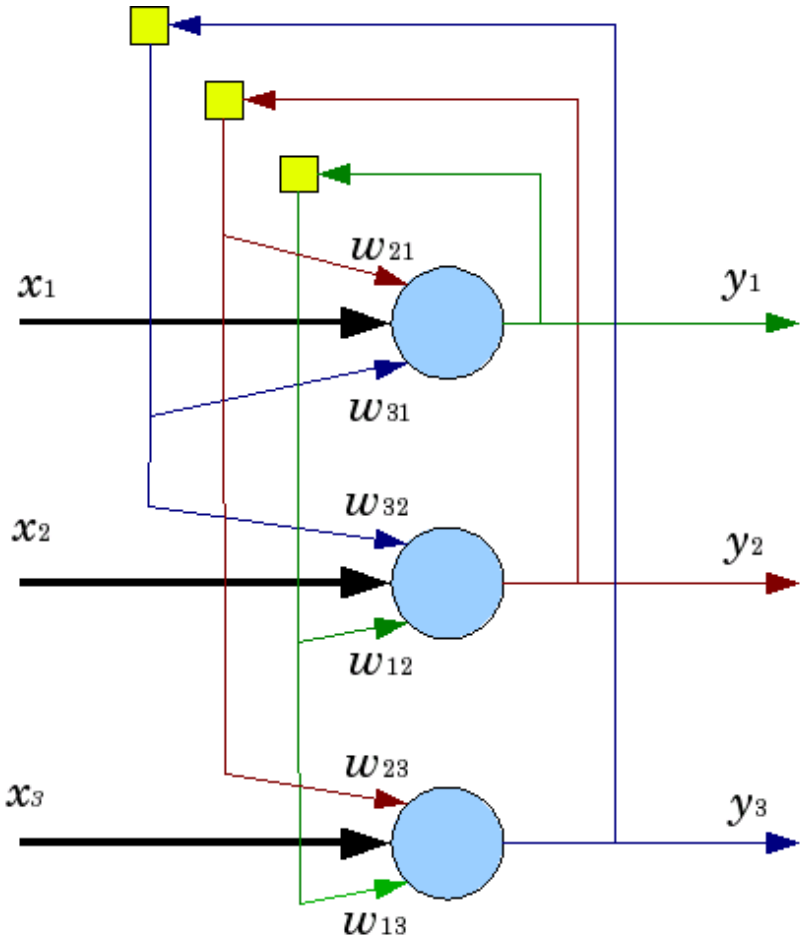


Рис 2.5. Обобщенная структурная схема сети Хопфилда

Размерности входных и **выходных сигналов** в сети ограничены при программной реализации только возможностями вычислительной системы, на которой моделируется нейронная сеть, при аппаратной реализации — технологическими возможностями. Размерности входных и выходных сигналов совпадают.

Каждый нейрон системы может принимать одно из двух состояний (что аналогично выходу нейрона с пороговой функцией активации) [23]:

$$x_i = \begin{cases} 1, \\ -1 \end{cases} \tag{2.8}$$

Благодаря своей биполярной природе нейроны сети Хопфилда иногда называют **спинами**.

Взаимодействие спинов сети описывается выражением («энергетической» функцией, которая уменьшается в процессе функционирования сети) [23]:

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}x_i x_j \qquad (2.9)$$

Где $\mathbf{w_{ij}}$ элемент матрицы взаимодействий **W**, которая состоит из весовых коэффициентов связей между нейронами. В эту матрицу в процессе обучения записывается **M** «образов» —**N**-мерных бинарных векторов: $S_m = (s_{m1}, s_{m2}, \ldots, s_{mN})$.

В сети Хопфилда **матрица связей** является симметричной ($w_{ij} = w_{ji}$), а диагональные элементы матрицы полагаются равными нулю ($w_{ii} = 0$), что исключает эффект воздействия нейрона на самого себя и является необходимым для сети Хопфилда.

Рассмотрим **общий алгоритм работы сети** [23]:

1. На стадии **инициализации сети** весовые коэффициенты **синапсов** (связей, по которым выходные сигналы одних нейронов поступают на входы других) устанавливаются следующим образом:

$$w_{ij} = \begin{cases} \sum_{k=0}^{m-1} x_i^k x_j^k, i \neq j \\ 0, \qquad i = j \end{cases} \qquad (2.10)$$

Где **i** и **j** ()– индексы, соответственно, предсинаптического (входного) и постсинаптического (выходного) нейронов; $\mathbf{w_{ij}}$ — i-й синаптический вес j-го нейрона; $\mathbf{x_i^k}$, $\mathbf{x_j^k}$ – i-ый и j-ый элементы вектора k-ого образца.

2. На входы сети подаётся неизвестный сигнал. Его ввод осуществляется непосредственной установкой значений:

$$\mathbf{y_j}(0) = x_j, \; j = 0...n-1 \qquad (2.11)$$

где $\mathbf{y_j}$ — **аксон** (т.е. выход) j-го нейрона. Поэтому обозначение на схеме сети входных синапсов в явном виде носит чисто условный характер; $\mathbf{y_j}(0)$ - нуль в скобке, означает нулевую итерацию в цикле работы сети.

3. Рассчитывается **новое состояние нейронов и новые значения аксонов**:

$$s_j(p+1) = \sum_{i=0}^{n-1} w_{ij} y_i(p) \quad (2.12)$$

где j=0...n-1,**p**– номер (конкретный шаг) итерации, - новое состояние нейрона;

$$y_j(p+1) = f\left[s_j(p+1)\right] \quad (2.13)$$

Где **f** – активационная функция в виде скачка, приведённая на рис. 2.6 б.

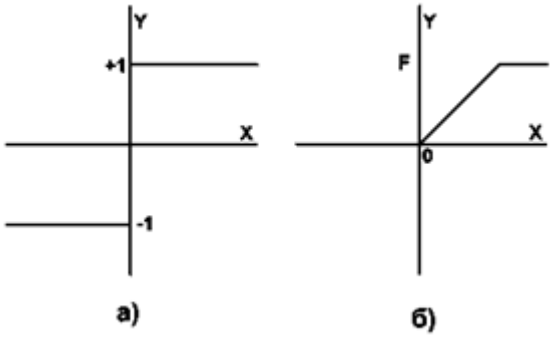


Рис. 2.6: а) жёсткая пороговая (передаточная) функция сети Хопфилда; б) активационная функция сети Хопфилда.

4. **Проверка**: изменились ли выходные значения аксонов за последнюю итерацию. Если да – переход к пункту 2, иначе (если выходы застабилизировались) – конец. При этом выходной вектор представляет собой образец, наилучшим образом сочетающийся с входными данными.

Алгоритм обучения сети Хопфилда имеет существенные отличия в сравнении с такими классическими алгоритмами обучения перцептронов, как метод коррекции ошибки или метод обратного распространения ошибки. Отличие заключается в том, что вместо последовательного приближения к нужному состоянию с вычислением ошибок, все коэффициенты матрицы рассчитываются по одной формуле, за один цикл, после чего сеть сразу готова к работе. Вычисление коэффициентов базируется на следующем

правиле: для всех заполненных образов X_i матрица связи должна удовлетворять уравнению [23]:

$$X_i^T = WX_i \tag{2.14}$$

Поскольку именно при этом условии состояния сети X_i будут устойчивы - попав в такое состояние, сеть в нём и останется.

Сеть Хопфилда содержит информацию о том, к каким классам нужно относить тот или иной образ, поэтому такую сеть можно отнести к **классу оптимизирующих сетей (фильтров)**. Отличительной особенностью фильтров является то, что матрица весовых коэффициентов настраивается детерминированным алгоритмом раз и навсегда, и затем весовые коэффициенты больше не изменяются [23].

В сети Хопфилда есть **обратные связи** и из-за этого нужно решать **проблему устойчивости**. Веса между нейронами в сети Хопфилда могут рассматриваться в виде матрицы взаимодействий W. Сеть с обратными связями является устойчивой, если её матрица симметрична и имеет нули на главной диагонали. В случае сети Хопфилда условие симметричности является необходимым, но не достаточным, в том смысле, что на достижение устойчивого состояния влияет ещё и **режим работы сети**. Ниже будет показано, что только асинхронный режим работы сети гарантирует достижение устойчивого состояния сети, в синхронном случае возможно бесконечное переключение между двумя разными состояниями (такая ситуация называется **динамическим аттрактором**, в то время как устойчивое состояние принято называть **статическим аттрактором**).

Расчёт весовых коэффициентов (**обучением сети)** происходит по следующей формуле [23]:

$$w_{ij} = \frac{1}{N} \sum_{d=1..m} X_{id} X_{jd} \tag{2.15}$$

где N - размерность векторов, m – число запоминаемых выходных векторов; d – номер запоминаемого выходного вектора; X_{ij} – i-я компонента

запоминаемого выходного j-го вектора.

Это выражение может стать более ясным, если заметить, что весовая матрица W может быть найдена вычислением внешнего произведения каждого запоминаемого вектора с самим собой и суммированием матриц, полученных таким образом. Это может быть записано в виде [23]:

$$W = \frac{1}{N} \sum_i X_i^T X_i \tag{2.16}$$

где X_i – i-й запоминаемый вектор-строка.

Как только веса заданы, сеть может быть использована для получения запомненного выходного вектора по данному входному вектору, который может быть частично неправильным или неполным. Для этого выходам сети сначала придают значения этого начального вектора. Затем сеть последовательно меняет свои состояния согласно формулам 3-его этапа алгоритма функционирования сети. Полученное устойчивое состояние y_j (статический аттрактор), или, возможно, в синхронном случае пара $\{y_j, y_{j+1}\}$ (динамический аттрактор), является ответом сети на данный входной образ.

Существуют **3 режима работы сети Хопфилда** [23]:

1) режим фильтрации;

2) синхронный режим;

3) асинхронный режим.

В **режиме фильтрации (восстановление повреждённых образов)**веса заданы, сеть может быть использована для получения запомненного выходного вектора по данному входному вектору, который может быть частично неправильным или неполным. Для этого выходам сети сначала придают значения этого начального вектора. Затем сеть последовательно меняет свои состояния согласно формуле нахождения общего алгоритма**.** Этот процесс называется **конвергенцией сети**.

Данный алгоритм можно описать так называемым локальным полем $\mathbf{a_i}$ действующим на нейрон x_i со стороны всех остальных нейронов сети [23]:

$$a_i(t) = \sum_{j=1, j \neq i}^{N} w_{ij} x_j(t-1) \quad (2.17)$$

После расчёта локального поля нейрона $a_i(t)$ это значение используется для расчёта значения выхода через функцию активации, которая в данном случае является пороговой (с нулевым порогом). Соответственно, значение выхода $x_i(t)$ нейрона i в текущий момент времени t рассчитывается по формуле [23]:

$$x_i(t) = sign(\sum_{j=1, j \neq i}^{N} w_{ij} x_j(t-1)) \quad (2.18)$$

Где $x_j(t-1)$ - значения выходов нейрона j в предыдущий момент времени.

Обычно ответом является такое устойчивое состояние, которое совпадает с одним из запомненных при обучении векторов, однако при некоторых условиях (в частности, при слишком большом количестве запомненных образов) результатом работы может стать так называемый **ложный аттрактор** ("химера"), состоящий из нескольких частей разных запомненных образов, а также в синхронном режиме сеть может прийти к динамическому аттрактору. Обе эти ситуации в общем случае являются нежелательными, поскольку не соответствуют ни одному запомненному вектору - а соответственно, не определяют класс, к которому сеть отнесла входной образ [23].

В **синхронном режиме работы,** если работа сети моделируется на одном процессоре, то при данном режиме последовательно просматриваются нейроны, однако их состояния запоминаются отдельно и не меняются до тех пор, пока не будут пройдены все нейроны сети. Когда все нейроны просмотрены, их состояния синхронно меняются на новые. Таким образом, достигается моделирование параллельной работы последовательным алгоритмом. При реально параллельном моделировании, этот режим фактически означает, что время передачи для каждой связи между элементами u_i и u_j одинаковое для каждой связи, что приводит к

параллельной работе всех связей, они одновременно меняют свои состояния, основываясь только на предыдущем моменте времени. Наличие таких синхронных тактов, которые можно легко выделить и приводит к пониманию синхронного режима. При синхронном режиме возможно бесконечное чередование двух состояний с разной энергией (динамический аттрактор). Поэтому синхронный режим практически для сети Хопфилда не используется.

В **асинхронном режиме работы сети,** если моделировать работу сети как последовательный алгоритм, то в данном режиме работы состояния нейронов в следующий момент времени меняются последовательно: вычисляется локальное поле для первого нейрона в момент t, определяется его реакция, и нейрон устанавливается в новое состояние (которое соответствует его выходу в момент t+1), потом вычисляется локальное поле для второго нейрона с учётом нового состояния первого, меняется состояние второго нейрона, и так далее - состояние каждого следующего нейрона вычисляется с учетом всех изменений состояний рассмотренных ранее нейронов. По сути при последовательной реализации сети Хопфилда явно не видно в чём заключается асинхронность, но это видно если сеть Хопфилда реализовать с **параллельными вычислениями.**

В асинхронном режиме невозможен динамический аттрактор - вне зависимости от количества запомненных образов и начального состояния, сеть непременно придет к устойчивому состоянию (статическому аттрактору).

Сеть Хопфилда может быть использована как ассоциативная память, для решения некоторых задач оптимизации, а также как фильтр (задачи распознавания образов) [23].

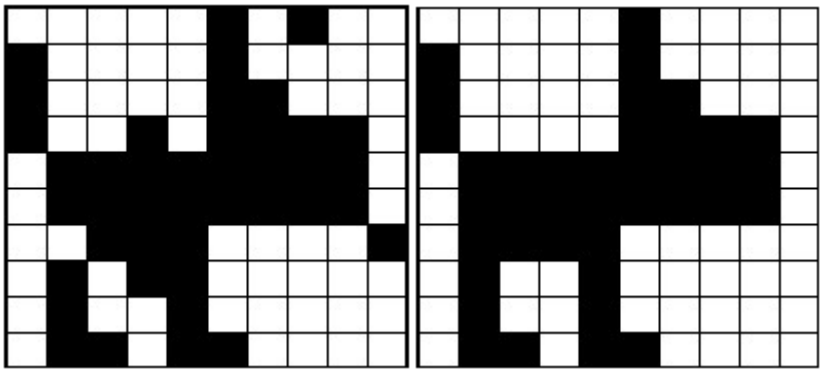
Чтобы организовать устойчивую **автоассоциативную память** с помощью данной сети с обратными связями, веса должны выбираться так, чтобы образовывать энергетические минимумы в нужных вершинах единичного гиперкуба.

На каждой итерации алгоритма функционирования сети понижается значение энергии нейронной сети. Это позволяет решать **комбинаторные задачи оптимизации**, если они могут быть сформулированы как задачи минимизации энергии.

Рассмотрим **пример восстановления повреждённого изображения** [23].

Если во время обучения сформировать матрицу весовых коэффициентов на основании эталонных бинарных векторов, то нейронная сеть в процессе работы будет менять состояния нейронов до тех пор, пока не перейдет к одному из устойчивых состояний.

Пусть имеется нейронная сеть размерностью **N=100**, в матрицу связей записан набор чёрно-белых картинок (-1 — чёрный цвет, +1 — белый), среди которых есть изображение собачки (рис. 2.7 б). Если установить начальное состояние сети близким к этому вектору (рис. 2.7 а), то в ходе динамики нейронная сеть восстановит исходное изображение (рис. 2.7 б). В этом смысле можно говорить о том, что сеть Хопфилда решает задачу распознавания образов (хотя строго говоря, полученное эталонное изображение ещё нужно превратить в номер класса, что в некоторых случаях может быть весьма вычислительно ёмкой задачей).



а) б)

Рис. 2.7: а) начальное состояние для сети Хопфилда; б) восстановленное изображение.

Достоинства, недостатки и модификации Сети хопфилда [23].

Достоинством сети Хопфилда является то, что она имеет огромное историческое значение. С этой модели началось возрождение интереса к нейронным сетям в середине 80-х годов. Также имеющиеся модификации применимы к решению современных задач области применения данной сети.

К сожалению, у нейронной сети Хопфилда есть **ряд недостатков** [23]:

1. Относительно небольшой объём памяти, величину которого можно оценить выражением:

$$M \approx \frac{N}{2log_2 N} \tag{2.19}$$

Попытка записи большего числа образов приводит к тому, что нейронная сеть перестаёт их распознавать.

2. Достижение устойчивого состояния не гарантирует правильный ответ сети. Это происходит из-за того, что сеть может сойтись к так называемым ложным аттракторам, иногда называемым "химерой" (как правило, химеры склеены из фрагментов различных образов).

3. При использовании коррелированных векторов-образцов возможно зацикливание сети в процессе функционирования.

4. Наряду с запомненными образами в сети хранятся и их негативы.

У сети Хопфилда существуют модификации. Одна из них предназначена для решения задач оптимизации, в частности задачи распределения работ между исполнителями.

Существует **модель сети Хопфилда с бинарными входными сигналами.**

Для увеличения ёмкости сети и повышения качества распознавания образов используют мультипликативные нейроны. Сети, состоящие из таких нейронов, называются **сетями высших порядков.**

2.3. Ассоциативная сеть Хемминга

Если в задаче ассоциативной памяти нет необходимости в том, чтобы нейросеть выдавала эталонный образец, а достаточно только номера образца, то для этих целей используется сеть Хемминга. Структурная схема сети Хемминга представлена на рисунке 2.8 [24].

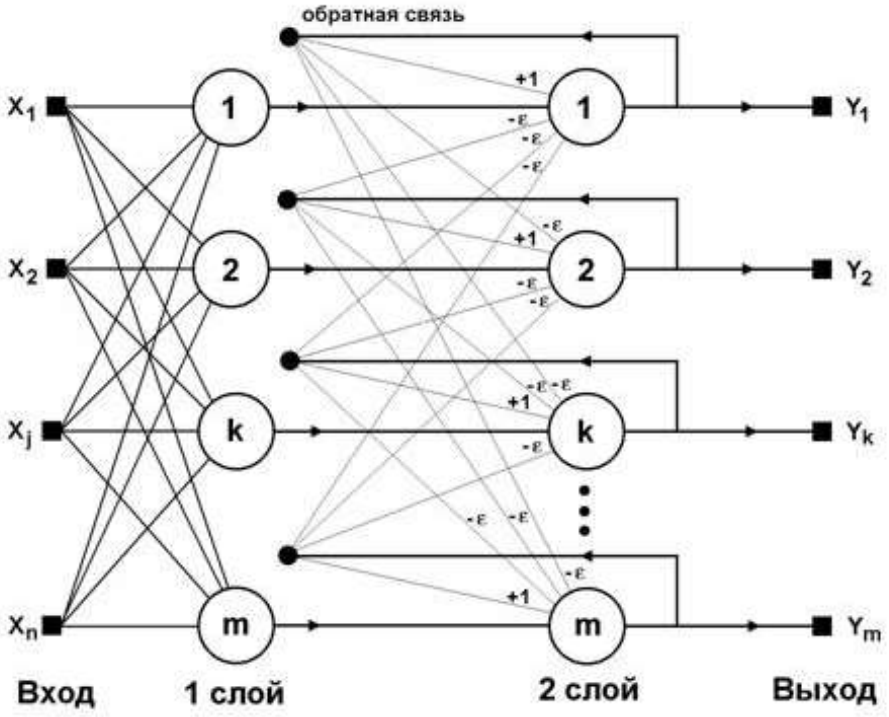


Рис. 2.8. Сеть Хемминга

Данная сеть, в сравнении с сетью Хопфилда, характеризуется меньшими вычислительными затратами. В сети Хемминга два слоя – первый и второй слои состоят из m нейронов и m равно числу образцов. Нейроны первого слоя имеют по n входных синапсов, где n - размерность входных векторов. Нейроны второго слоя связаны между собой обратными, отрицательными связями. Обратная связь от аксона на владельца нейрона равен +1. Суть работы состоит в нахождении расстояния Хемминга от тестируемого образца до всех образцов. Расстоянием Хемминга называется число отличающихся битов в двух бинарных векторах.

 - расстояние Хемминга равно 0.

 - расстояние Хемминга равно 2.

Сеть должна выбрать образец с минимальным расстоянием Хемминга до поданного входного сигнала – в результате активируется один выход, отвечающий за данный эталонный образец.

При инициализации сети весовым коэффициентам первого слоя и порогу активационной функции присваиваются следующие значения [24]:

$$w_{ik} = \frac{x_i^k}{2},\ i=0...n-1,\ k=0...m-1 \qquad (2.20)$$
$$T_k = n / 2,\ k = 0...m-1$$

Где x_i – i-ый элемент k-ого образца.

Весовые коэффициенты тормозящих синапсов во втором слое берут равными некоторой величине 0 < □ < 1/m. Синапс нейрона, связанный с его же аксоном имеет вес +1.

Алгоритм работы сети Хэмминга следующий [24]:

1. На входы сети подается неизвестный вектор **X** = {x_i:i=0...n-1}, исходя из которого рассчитываются состояния нейронов первого слоя (верхний индекс в скобках указывает номер слоя):

$$y_j^{(1)} = s_j^{(1)} = \sum_{i=0}^{n-1} w_{ij} x_i + T_j,\ j=0...m-1 \qquad (2.21)$$

После этого полученными значениями инициализируются значения аксонов второго слоя:

$$y_j^{(2)} = y_j^{(1)},\ j = 0...m-1 \qquad (2.22)$$

2. Вычислить новые состояния нейронов второго слоя:

$$s_j^{(2)}(p+1) = y_j(p) - \varepsilon \sum_{k=0}^{m-1} y_k^{(2)}(p), k \neq j, j = 0 \ldots m-1 \qquad (2.23)$$

и значения их аксонов:

$$y_j^{(2)}(p+1) = f\left[s_j^{(2)}(p+1)\right], j = 0 \ldots m-1 \qquad (2.24)$$

Активационная функция f имеет вид порога, причем величина F

должна быть достаточно большой, чтобы любые возможные значения аргумента не приводили к насыщению.

3. Проверить, изменились ли выходы нейронов второго слоя за последнюю итерацию. Если да – перейди к шагу 2. Иначе – завершение работы.

Из оценки алгоритма видно [24], что роль первого слоя нейронов весьма условна: воспользовавшись один раз на шаге 1 значениями его весовых коэффициентов, сеть больше не обращается к нему, поэтому первый слой может быть вообще исключен из сети (просто заменен на матрицу весовых коэффициентов.

2.4. Сети встречного распространения

В сети встречного распространения (рис 2.9) объединены два алгоритма: самоорганизующаяся карта Кохонена и звезда Гроссберга [18]. Их объединение ведет к свойствам, которых нет ни у одного из них в отдельности [25].

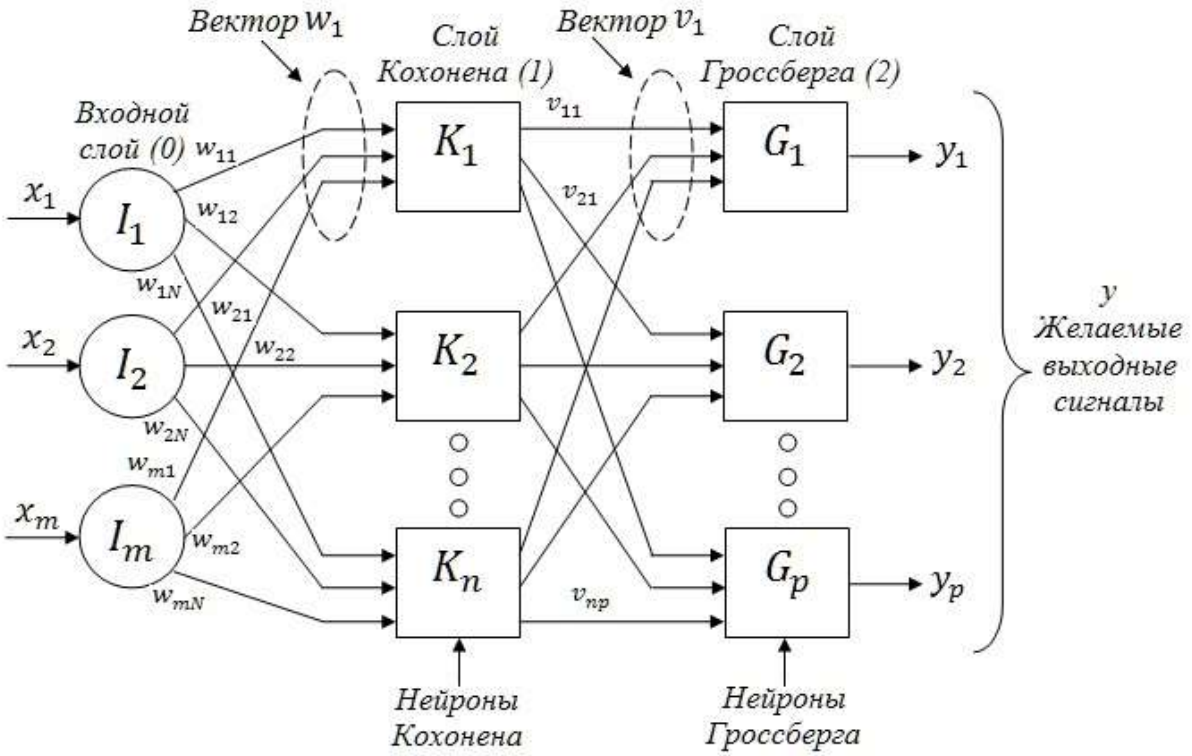


Рис.2.9. Сеть с встречным распространением без обратных связей

Сеть Кохонена – это двухслойная нейронная сеть, содержащая *входной слой* (слой входных нейронов) и *слой Кохонена* (слой активных нейронов). Слой Кохонена может быть: одномерным, двумерным или трехмерным [18].

В первом случае активные нейроны расположены в цепочку. Во втором случае они образуют двухмерную сетку (обычно в форме квадрата или прямоугольника), а в третьем случае они образуют трехмерную конструкцию в виде куба или параллелепипеда.

В силу отсутствия обучающей последовательности образов в виде звука или изображений, для каждого из которых известна от учителя принадлежность к тому или иному классу, определение весов нейронов слоя Кохонена основано на использовании алгоритмов классической классификации (кластеризации или самообучения) [18].

При обучении слоя Кохонена на вход подается входной вектор и вычисляются его скалярные произведения с векторами весов, связанными со всеми нейронами Кохонена. Нейрон с максимальным значением скалярного произведения объявляется «победителем» и его веса подстраиваются. Так как скалярное произведение, используемое для вычисления величин S(сумма входов нейрона), является мерой сходства между входным вектором и вектором весов, то процесс обучения состоит в выборе нейрона Кохонена с весовым вектором, наиболее близким к входному вектору, и дальнейшем приближении весового вектора к входному. Сеть самоорганизуется таким образом, что данный нейрон Кохонена имеет максимальный выход для данного входного вектора. Уравнение, описывающее процесс обучения имеет следующий вид:

$$w_н = w_с + \alpha(x - w_с) \qquad (2.25)$$

где $w_н$ – новое значение веса, соединяющего входную компоненту x с выигравшим нейроном; $w_с$ – предыдущее значение этого веса; α –

коэффициент скорости обучения, который может варьироваться в процессе обучения.

Каждый вес, связанный с выигравшим нейроном Кохонена, изменяется пропорционально разности между его величиной и величиной входа, к которому он присоединен. Направление изменения минимизирует разность между весом и его входом.

Слой Гроссберга обучается относительно просто. Входной вектор, являющийся выходом слоя Кохонена, подается на слой нейронов Гроссберга, и выходы слоя Гроссберга вычисляются, как при нормальном функционировании. Далее, каждый вес корректируется лишь в том случае, если он соединен с нейроном Кохонена, имеющим ненулевой выход. Величина коррекции веса пропорциональна разности между весом и требуемым выходом нейрона Гроссберга, с которым он соединен. Это выражается следующей формулой [25]:

$$v_{ijн} = v_{ijc} + \beta(y_j - v_{ijc})k_i \qquad (2.26)$$

где k_i – выход i-го нейрона Кохонена (только для одного нейрона Кохонена он отличен от нуля); y_j – j-ая компонента вектора желаемых выходов.

Первоначально β берется равным ~0,1 и затем постепенно уменьшается в процессе обучения.

Таким образом, веса слоя Гроссберга будут сходиться к средним величинам от желаемых выходов, тогда как веса слоя Кохонена обучаются на средних значениях входов. Обучение слоя Гроссберга – это обучение с учителем, алгоритм располагает желаемым выходом, по которому он обучается. Обучающийся без учителя, самоорганизующийся слой Кохонена дает выходы в недетерминированных позициях. Они отображаются в желаемые выходы слоем Гроссберга [25].

Сети с встречным распространением позволяют сжимать данные перед их передачей, уменьшая тем самым число битов, которые должны быть

переданы. Так, если требуется передать некоторое изображение, то оно может быть разбито на подизображения *S*, как показано на рис. 2.10. Каждое подизображение разбито на пиксели. Тогда каждое подизображение является вектором, элементами которого являются пиксели, из которых состоит подизображение. При этом можно допустить, что каждый пиксель – это единица (если он светлый) или нуль (если он черный). Если в подизображении имеется *n* пикселей, то для его передачи потребуется n бит. При этом если допустимы некоторые искажения изображения, то для его передачи требуется существенно меньшее число битов, что позволяет передавать изображение быстрее. Это возможно из-за статистического распределения векторов подизображений, поскольку некоторые из них могут встречаться часто, тогда как другие, встречаться так редко, что могут быть аппроксимированы довольно грубо. Таким, образом, метод, называемый векторным квантованием, находит более короткие последовательности битов, наилучшим образом представляющие эти подизображения [27].

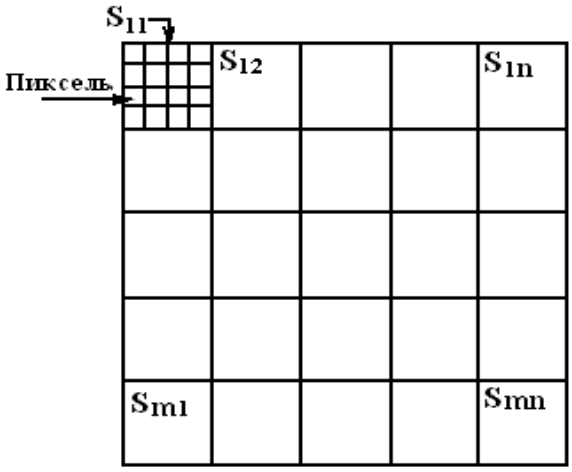


Рис.2.10. Пример разбиения изображения на пиксели

Сеть встречного распространения может быть использована для выполнения векторного квантования. Множество векторов подизображений используется в качестве входа для обучения слоя Кохонена по методу аккредитации (алгоритм обучения, в котором для каждого входного вектора активировался лишь один нейрон Кохонена), когда выход единственного

нейрона равен 1. Веса слоя Гроссберга обучаются выдавать бинарный код номера того нейрона Кохонена, выход которого равен 1. Например, пусть дана нейронная сеть с общим количеством 50 нейронов, количество выходных равно 10, если выходной сигнал нейрона №7 равен 1 (а все остальные равны 0), то слой Гроссберга будет обучаться выдавать двоичный код числа 7 (00…000111). Это и будет являться более короткой битовой последовательностью передаваемых символов.

На приемном конце идентичным образом обученная сеть встречного распространения принимает двоичный код и реализует обратную функцию, аппроксимирующую первоначальное подизображение.

Этот метод применялся на практике как к речи, так и к изображениям, с коэффициентом сжатия данных от 10:1 до 100:1. При этом, как указывается в литературе [27] качество было приемлемым, хотя некоторые искажения данных на приемном конце признаются неизбежными.

2.5. Сверточные нейронные сети

Свёрточная нейронная сеть (англ. *convolutional neural network, CNN*) — специальная архитектура искусственных нейронных сетей, предложенная Яном Лекуном и нацеленная на эффективное распознавание изображений, входит в состав технологий глубокого обучения. Использует некоторые особенности зрительной коры, в которой были открыты так называемые простые клетки, реагирующие на прямые линии под разными углами, и сложные клетки, реакция которых связана с активацией определённого набора простых клеток. Таким образом, идея свёрточных нейронных сетей заключается в чередовании свёрточных слоев и субдискретизирующих слоев [29].

Сверточные нейронные сети обьединяют три архитектурных идеи для обеспечения частичной устойчивости к изменению масштаба, повороту, сдвигу и пространственным искажениям [28]:

1. Локальные рецепторные поля (позволяют учитывать двумерную топологию входных данных);
2. Общие веса (обеспечивают детектирование общих черт в любом месте изображения и уменьшают количество настраиваемых параметров);
3. Иерархическая организация с пространственными подвыборками (позволяет строить иерархии признаков).

Для обучения сверточных нейронных сетей может применяться как стандартный метод обратного распространения ошибки, так и его различные модификации [30].

Архитектура свёрточной нейронной сети [29].

В перцептроне, который представляет собой полносвязную нейронную сеть, каждый нейрон связан со всеми нейронами предыдущего слоя, причем каждая связь имеет свой персональный весовой коэффициент (рис 2.11) [29]. В свёрточной нейронной сети в операции свёртки используется лишь ограниченная матрица весов небольшого размера, которую «двигают» по всему обрабатываемому слою, формируя после каждого сдвига сигнал активации для нейрона следующего слоя с аналогичной позицией. Матрица весов, которую также называют набором весов или *ядром свёртки*, построена таким образом, что графически кодирует какой-либо один признак, например, наличие наклонной линии под определенным углом. Тогда следующий слой, получившийся в результате операции свёртки такой матрицей весов, показывает наличие данной наклонной линии в обрабатываемом слое и её координаты, формируя так называемую карту признаков. В результате получаем не один набор весов свёрточной нейронной сети, а целую гамму, кодирующую всевозможные линии и дуги под разными углами. Проход каждым набором весов формирует свой собственный экземпляр карты признаков, делая нейронную сеть многомерной (много независимых карт признаков на одном слое). Также следует отметить, что при переборе слоя матрицей весов её передвигают

обычно не на полный шаг (размер этой матрицы), а на небольшое расстояние. Так, например, при размерности матрицы весов 5×5 её сдвигают на один или два нейрона (пикселя) вместо пяти, чтобы не «перешагнуть» искомый признак.

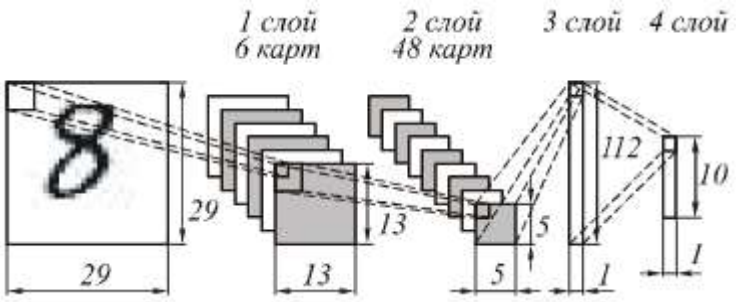


Рис 2.11. Пример построения сверточной нейронной сети для распознавания рукописных символов.

Операция субдискретизации (также называемая как «операция подвыборки»), выполняет уменьшение размерности сформированных карт признаков. В данной архитектуре сети считается, что информация о факте наличия искомого признака важнее точного знания его координат, поэтому из нескольких соседних нейронов карты признаков выбирается максимальный и принимается за один нейрон карты признаков уменьшенной размерности. Также иногда применяют операцию нахождения среднего между соседними нейронами. За счёт данной операции, помимо ускорения дальнейших вычислений, сеть становится более инвариантной к масштабу входного изображения.

Таким образом, повторяя друг за другом, несколько слоёв свёртки и субдискретизации строится свёрточная нейронная сеть. Чередование слоёв позволяет составлять карты признаков из карт признаков, что на практике означает способность распознавания сложных иерархий признаков. Обычно после прохождения нескольких слоев карта признаков вырождается в вектор или даже скаляр, но таких карт признаков становится сотни. На выходе сети часто дополнительно устанавливают несколько слоев полносвязной

нейронной сети (перцептрон), на вход которому подаются оконечные карты признаков [29].

Если на первом слое ядро свёртки проходит только по одному исходному изображению, то на внутренних слоях одно и то же ядро проходит параллельно по всем картам признаков этого слоя, а результат свертки суммируется, формируя (после прохождения функции активации) одну карту признаков следующего слоя, соответствующую этому ядру свертки.

Основной проблемой развития нейронных сетей являются высокие вычислительные затраты и для решения этой проблемы на данный момент используются специальные программные обеспечения для организации параллельных и распределённых вычислений, такие как нейрочипы, нейропроцессоры, ПЛИС, распределённые кластерные системы, GRID-технологии [31].

Аппаратно-программный комплекс CUDA (Compute Unified Device Architecture) также позволяет использовать процессоры видеокарт как ускорители научных и инженерных расчетов и проводить вычисления по эффективности сравнимые с кластерными системами [32]. Особенностью оборудования, поддерживающего технологию CUDA, является возможность обеспечивать на порядок большую (по сравнению с кластерами) пропускную способность при работе с памятью [28].

В среде NVIDIA CUDA на примере сверточной нейронной сети были проведены ряд экспериментов по распознаванию образов изображений, результаты которой показали, что длительность обучения нейронной сети на видеоадаптере уменьшена в 5,96, а распознавания набора тестовых образцов – в 8,76 раза по сравнению с оптимизированным алгоритмом, выполняющим вычисления только на центральном процессоре (CPU). Показана перспективность реализации нейросетевых алгоритмов на графических процессорах [28].

Как говорилось ранее, основным недостатком нейронных сетей является

высокие вычислительные затраты, которые в данной реализации были устранены путем замены в среде CUDA циклов параллельно выполняющимися командами вида SIMD [28].

Выводы

В результате анализа нейросетевых технологий было установлено, что основными идеями, из которых возникли и выросли нейросети и нейромоделирование, являются следующие [33]:

- Нейронная сеть *имитирует структуру и свойства нервной системы* живых организмов.
- Каждый нейрон выполняет небольшой объем работ − например, суммирует пришедшие на него сигналы с некоторыми весовыми коэффициентами и дополнительно нелинейно преобразует эту взвешенную сумму входных данных. Другим распространённым вариантом является нейрон-детектор, выдающий высокий выходной сигнал при малых отличиях своих входных сигналов от некоторого запомненного эталона, и низкий выходной сигнал при существенных отличиях.
- Нейроны группируются в последовательность слоёв. Но бывают и рекуррентные структуры, обеспечивающие циркуляцию некоторого набора внутренних сигналов.
- Нейроны, составляющие некоторый слой сети, работают в параллельном режиме.
- Процесс работы нейросети представляет собой движение потока внешних сенсорных данных (от некоторого "входа" к "выходу") и преобразование этих данных. В общем случае поток данных (сигналов) может формировать и перекрёстные, и обратные связи.

- Искусственная нейросеть, *может обучаться*: она содержит внутренние адаптивные параметры нейронов и своей структуры, и, меняя их, может менять свое поведение, добиваясь улучшения точности решения некоторой задачи.
- Место программирования занимает *обучение нейронной сети*.
- Кроме обучения с учителем (на основе знаний об известных эталонных ответах для некоторого набора ситуаций) возможно и обучение без учителя − при этом происходит анализ описаний ситуаций.
- *Нейронная сеть способна обучаться решению задач, для которых у человека не существует формализованных, быстрых или работающих с приемлемой точностью теоретических или эмпирических алгоритмов*.
- *Структура нейросети может быть адаптирована к решаемой задаче*.

3. ПРИМЕНЕНИЕ НЕЙРОТЕХНОЛОГИЙ ДЛЯ ОБРАБОТКИ ИЗОБРАЖЕНИЙ И ОЦЕНКА ИХ ЭФФЕКТИВНОСТИ

3.1. Общие положения

С самого начала развития компьютерной техники были намечены два принципиально разных подхода к обработке информации: принцип последовательной обработки сигналов и параллельное распознавание образов. Последовательная обработка была реализована в виде общераспространенных процессоров электронно-вычислительных машин, что определило основное применение ЭВМ на десятилетия вперед - решение задач с помощью запрограммированных человеком алгоритмов [34].

В то же время описание операций над многобитовыми образами при использовании последовательного принципа обработки команд оказалось очень трудной задачей из-за большой сложности описания образов. Требовались качественно другие подходы и модели обработки информации. За основу новых решений были приняты принципы функционирования биологических нейронных сетей, которые составляют основу деятельности человеческого мозга с «образами» внешнего мира - распознавание сенсорной информации и выработка адекватной реакции на внешние воздействия. Таким образом, появились аналоги биологических нейронных сетей в виде искусственных нейросетей, реализуемых на компьютерах. Основная задача нейросетей - не выполнение внешних алгоритмов, а выработка собственных в процессе обучения и отбраковки неверных решений, т.е. устранения ошибок каждого нейрона [34].

Нейрокомпьютеры позволяют с высокой эффективностью решать целый ряд интеллектуальных задач. Это задачи распознавания образов, адаптивного управления, прогнозирования, диагностики и т.д. [35].

Нейрокомпьютеры отличаются от ПК предыдущих поколений не просто большими возможностями. Принципиально меняется способ использования

машины, где место программирования занимает обучение, а нейрокомпьютер учится решать задачи.

Таким образом, отличия нейрокомпьютеров от вычислительных устройств предыдущих поколений заключается в следующем[35]:

1. параллельная работа очень большого числа простых вычислительных устройств обеспечивает огромное быстродействие;
2. нейронная сеть способна к обучению, которое осуществляется путем настройки параметров сети;
3. высокая помехо- и отказоустойчивость нейронных сетей;
4. простое строение отдельных нейронов позволяет использовать новые физические принципы обработки информации для аппаратных реализаций нейронных сетей.

В основе построения нейрокомпьютеров данного типа лежит использование ПЦОС или ПЛИС, объединенных между собой в соответствии с архитектурой, которая обеспечивает параллельность выполнения вычислительных операций [18].

Как правило, такие нейрокомпьютеры строятся на основе гибкой модульной архитектуры, которая обеспечивает простоту конфигурации системы и наращиваемость вычислительной мощности путем увеличения числа процессорных модулей или применения более производительных ПЦОС (рис.4.1). Системы реализуются в основном на базе несущих модулей стандартов ISA, PCI, VME.

Основные функциональные элементы данных нейрокомпьютеров [18]:

- модуль матричных ПЦОС,
- рабочая память,
- память программ,
- модуль обеспечения ввода/вывода сигналов (включающий АЦП, ЦАП и TTL линии),
- модуль управления, который может быть реализован на основе специализированного управляющего ПЦОС (УП), на основе

ПЛИС или иметь распределенную структуру, при которой функции общего управления распределены между матричными ПЦОС.

Рис 3.1. Обобщенная функциональная схема нейрокомпьютера, реализованного на основе ПЦОС или (и) ПЛИС.

Реализация нейрокомпьютеров и специализированных вычислителей с массовым параллелизмом на базе ПЦОС и ПЛИС является весьма эффективным решением задач цифровой обработки сигналов, обработки видео- и аудиоданных и построения технических систем управления.

При создании нейрокомпьютеров используется гибридная схема в которой блок матричных вычислений реализуется на базе кластерного соединения ПЦОС, а логика управления на основе ПЛИС. В качестве элементной базы матричного кластера обычно используются ПЦОС ADSP21060 и TMS320C44, но ближайшее время им на смену придут ADSP2106x и TMS320C67xx. По оценке специалистов [18], в ближайшем будущем матричное ядро чаще будет реализовываться на базе нейропроцессоров, а ПЦОС и ПЛИС останутся основой для построения логики управления (например, Synapse 3).

Для построения нейрокомпьютеров данного типа наиболее перспективным является использование сигнальных процессоров с плавающей точкой ADSP2106x, TMS320C4x,8x, DSP96002 и др.

3.2.Анализ элементной базы нейропроцессоров для построения нейросетей

Нейропроцессор – это кристалл, который обеспечивает выполнение нейросетевых алгоритмов в реальном масштабе времени [18].

Среди разновидностей кристаллов, используемых в качестве нейропроцессоров выделим следующие (рис. 3.2) [18]:

- специализированные нейрочипы;
- заказные кристаллы (ASIC);
- встраиваемые микроконтроллеры (mC);
- процессоры общего назначения (GPP);
- перепрограммируемые логические интегральные схемы (FPGA, ПЛИС);
- процессоры цифровой обработки сигналов (ПЦОС);
- транспьютеры.

Специализированные нейрочипы часто реализуются на основе процессорных матриц (систолических процессоров). Такие нейрочипы близки к обычным RISC-процессорам, но объединяют в своем составе некоторое число процессорных элементов. При этом управляющая и дополнительная логика, как правило, строится на базе дополнительных схем.

Различают также нейросигнальные процессоры, ядро которых представляет собой типовой ПЦОС, а реализованная на кристалле дополнительная логика обеспечивает выполнение характерных нейросетевых операций (например, дополнительный векторный процессор и т.п.).

Транспьютеры (англ. *transputer* от слов «транзистор» и «компьютер») — элемент построения многопроцессорных систем, выполненный на одном

кристалле большой интегральной схемы [42]. В частности T414, T800, T9000, и транспьютероподобные элементы являются важным компонентом ВСМП (вычислительные системы с массовым параллелизмом), однако, их применение сдвигается в сторону коммутационных систем и сетей ЭВМ [18].

Рис 3.2. Кристаллы, используемые в качестве нейропроцессоров.

Для оценки производительности устройств (реализованных на основе ПЦОС и ПЛИС), применяемых для ЦОС, контролируется время выполнения типовых операций ЦОС, таких как цифровая фильтрация, БПФ и др. В свою очередь, для оценки производительности нейропроцессоров и

нейрокомпьютеров применяется ряд специальных показателей (параметров) [18]:

- MAC (Multiply-Accumulate) – миллионов умножений с накоплением в секунду;
- CUPS (Connections Update per Second) – число измененных значений весов в секунду (оценивает скорость обучения);
- CPS (Connections per Second) – число соединений (умножений с накоплением) в секунду (оценивает производительность);
- CPSPW = CPS/Nw, где Nw – число синапсов в нейроне;
- CPPS – число соединений примитивов в секунду [18]:

$$CPPS = CPS \bullet Bw \bullet Bs, \quad (3.1)$$

Где: Bw, Bs – разрядность чисел, отведенных под веса и синапсы.

Ориентация процессоров на выполнение нейросетевых операций обуславливает, с одной стороны, повышение скоростей обмена между памятью и параллельными арифметическими устройствами, а с другой стороны, уменьшение времени весового суммирования (умножения и накопления) за счет применения фиксированного набора команд типа регистр-регистр.

Примером построения нейропроцессора может быть нейропроцессор производства НТЦ «Модуль» **Л1879ВМ3** [37].

Структурная схема СНК (системы на кристалле) приведена на рис 3.3 [37]. Микросхема работает от внешнего синусоидального источника синхросигнала с частотой 600 МГц. Блок синхронизации формирует 300 МГц сигналы для управления АЦП и ЦАП и 150 МГц сигналы для управления цифровой частью микросхемы. Как и в любой СНК в микросхеме 1879МВ3 часть вычислительных функций выполняется аппаратно, для этого предусмотрен набор функциональных устройств, объединенных в аналоговый интерфейс. Помимо двух АЦП и 4-ёх ЦАП, в него входят: генератор случайного шума; таймер реального времени (24-разрядный

счетчик с частотой тактирования 150 МГц); арифметические устройства, а также набор управляющих и накопительных регистров.

Контроллер CU (Control Unit) предназначен для формирования адресов команд, их считывания и дешифрации, а также для формирования сигналов управления всеми исполнительными узлами СНК.

Основные виды команд контроллера [37]: операции безусловного перехода, вызов подпрограммы и возврата из неё, команда перехода в режим ожидания, операции загрузки констант в регистры, межрегистровые пересылки. Кроме того, поддерживаются команды управления масками прерываний, счётчиками событий и задержек сигнальных каналов и т.д.

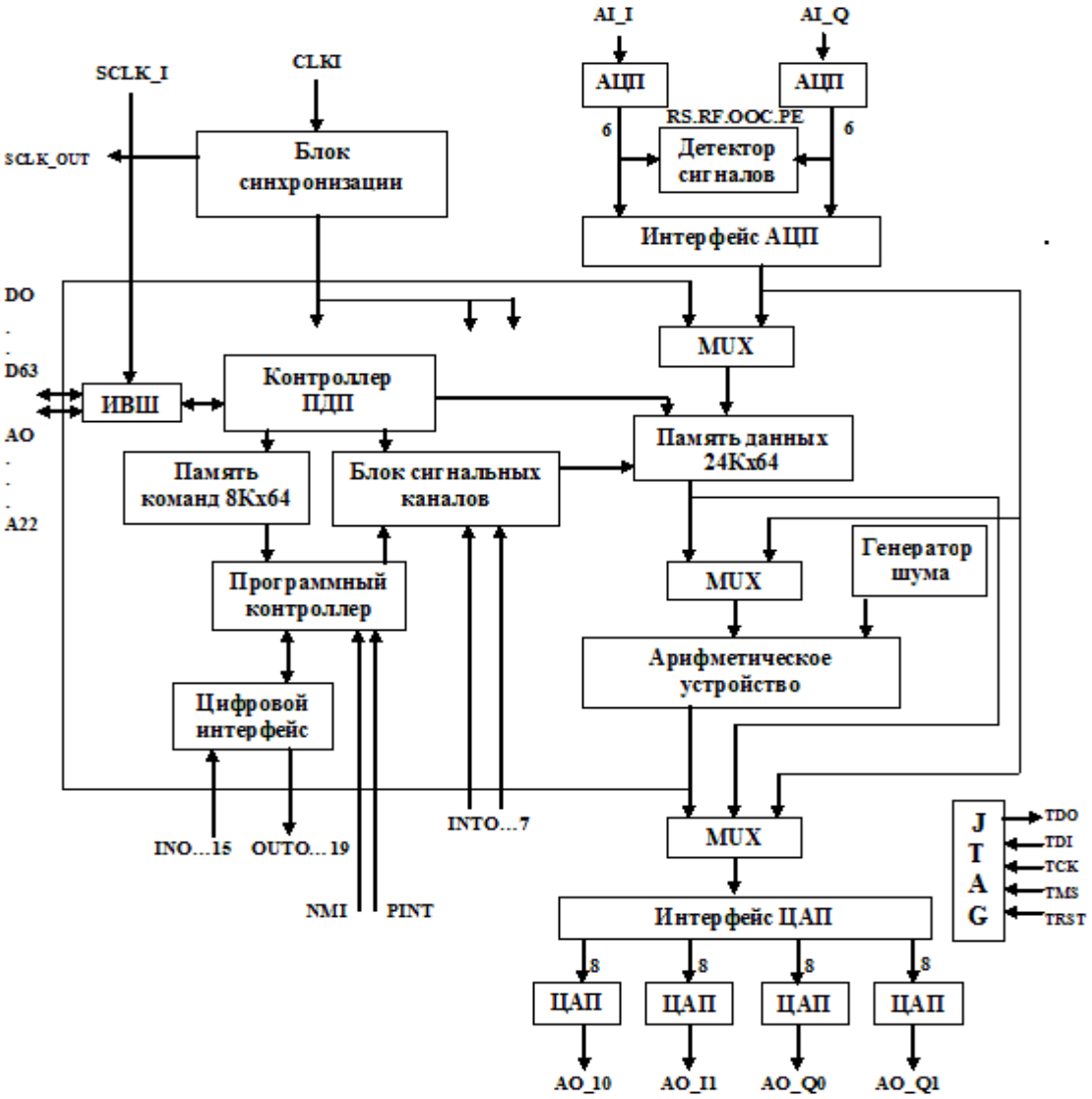


Рис 3.3. Структурная схема нейропроцессора 1879ВМ3

Внутренняя память СБИС 1879ВМ3 объемом 2Мбит позволяет ей принимать и сохранять высокочастотные аналоговые сигналы. Большое количество программируемых счетчиков и развитая система внутренних и внешних прерываний обеспечивают выдачу на аналоговые выходы однократных и периодических высокочастотных сигналов, хранящихся во внутренней памяти контроллера, в реальном масштабе времени с требуемыми задержками и длительностями. Встроенные быстродействующие арифметические узлы обеспечивают программируемое усиление входных сигналов, их суммирование с выходными сигналами, программируемое изменение сдвига частоты выходных сигналов. Внешняя 64-разрядная шина обеспечивает быстрый обмен командами и данными с внешней памятью или ЦПС как в режиме прямого доступа к памяти (ПДП), так и в режиме произвольного доступа ЦПС к внутренней памяти контроллера [37].

На рисунке 3.4 показана инструментальная плата с СНК 1879ВМ3 для подключения к персональному компьютеру.

Рис. 3.4. Инструментальная плата с нейропроцессором 1879ВМ3.

3.3. Анализ программного обеспечения для работы нейронных сетей

При построении нейронных сетей используется специальное программное обеспечение, основное на поддержке нейротехнологий. Под ***Нейротехнологией*** понимаются технологии обработки информации на базе моделей, методов, алгоритмов, программ, моделирующих или имитирующих работу нейронных сетей и процессы решения задач *искусственного интеллекта*. Такие технологии позволяют эффективно реализовывать такие свойства нейросетей, как *параллелизм, самообучение, распознавании, адаптивность и перестройку структуры* [38].

Поскольку нейрокомпьютеры представляют собой весьма дорогие устройства, то на практике часто производится моделирование работы нейросети на обычном ПК с помощью специального программного обеспечения. При этом, программу моделирования нейронной сети обычно называют программой-имитатором или нейропакетом, понимая под этим программную оболочку, эмулирующую для пользователя среду нейрокомпьютера на обычном компьютере [39].

В настоящее время на рынке программного обеспечения имеется множество самых разнообразных программ для моделирования нейронных сетей. Несмотря на большое разнообразие таких программ, в них можно выделить несколько основных функций, которые реализованы во всех этих программах:

Формирование (создание) нейронной сети [39]

Для решения разных практических задач требуются различные модели нейронных сетей. Модель нейронной сети определяется моделями нейронов и структурой связей сети.

Программы-имитаторы в зависимости от структуры связей реализуют следующие группы нейронных сетей:

- *Многослойные нейронные сети*. Нейроны в таких сетях делятся на группы с общим входным сигналом - слои. Различают несколько типов связей между слоями с номерами q и (q + p):
  - последовательные (p=1);
  - прямые (p > 1);
  - обратные (p < 0).
- Связи между нейронами одного слоя называют латеральными (боковыми).
- *Полносвязные нейронные сети*. Каждый нейрон в полносвязных сетях связан со всеми остальными. На каждом такте функционирования сети на входы нейронов подается внешний входной сигнал и выходы нейронов предыдущего такта.
- *Нейронные сети с локальными связями*. Нейроны в таких сетях располагаются в узлах прямоугольной или гексагональной решетки. Каждый нейрон связан с небольшим числом (4, 6 или 8) своих топологических соседей.
- *Неструктурированные нейронные сети*. К этой группе относятся все модели нейронных сетей, которые нельзя отнести ни к одной из предыдущих групп.

Модели реализуемых программами-имитаторами нейронов чрезвычайно разнообразны. В простейшем случае нейроны первого порядка выполняет взвешенное суммирование компонентов входного вектора и нелинейное преобразование результата суммирования. Нейроны более высоких порядков осуществляют перемножение двумерных матриц и многомерных тензоров (скаляры, векторы, билинейные формы).

В моделях нейронов используются различные варианты нелинейных преобразований. Наиболее часто используются сигмоидальные, кусочно-линейные и жесткие пороговые функции активации. В сети все нейроны могут иметь как одинаковые (гомогенная сеть), так и различные функции активации (гетерогенная сеть).

Для построения нейронной сети, ориентированной на решение конкретной задачи, используются процедуры формирования нейронных сетей, которые обеспечивают ввод указанных характеристик моделей нейронов и структур нейронных сетей.

Обучение нейронной сети [39]

В большинстве программ-имитаторов предлагаются стандартные процедуры обучения нейронных сетей, ориентированные на конкретные нейропарадигмы.

Как правило, в нейропакетах реализуется возможность задания различных типов данных и различных размерностей входных и выходных сигналов в зависимости от решаемой задачи. В качестве входных данных в обучающей выборке могут использоваться растровые изображения, таблицы чисел, распределения. Типами входных данных являются [39]:

- бинарные (0 и 1):
- биполярные числа (-1 и +1):
- целые или действительные числа из некоторого диапазона

Выходные сигналы сети являются векторы целых или действительных чисел.

Для решения практических задач часто требуются обучающие выборки большого объема. Поэтому в ряде нейропакетов предусмотрены средства, облегчающие процесс формирования и использования обучающих примеров. Однако в настоящее время отсутствует универсальная методика построения обучающих выборок и поэтому набор обучающих примеров, как правило, формируется индивидуально для каждой решаемой задачи.

В качестве функции ошибки, численно определяющей сходство всех текущих выходных сигналов сети и соответствующих требуемых выходных сигналов обучающей выборки, в большинстве случаев используется среднеквадратичное отклонение. Однако в ряде нейроимитаторов существует либо возможность выбора, либо задания своей функции ошибки.

Реализуемые в нейропакетах алгоритмы обучения нейронных сетей можно разделить на три группы: градиентные, стохастические, генетические:

- **градиентные алгоритмы** (первого и второго порядков) основаны на вычислении частных производных функции ошибки по параметрам сети;
- **стохастические алгоритмы** организуют поиск минимума функции ошибки случайным образом;
- **генетические алгоритмы** комбинируют свойства стохастических и градиентных алгоритмов, то есть на основе аналога генетического наследования реализуют перебор вариантов, а на основе аналога естественного отбора - градиентный спуск (нахождение локального экстремума функции с помощью движения вдоль градиента).

При обучении нейронных сетей, как правило, используются следующие критерии:

- при достижении некоторого малого значения функции ошибки;
- в случае успешного решения всех примеров обучающей выборки (при неизменности выходных сигналов сети).

В нейроимитаторах предусмотрено наличие специальных процедур инициализации перед обучением сети, т. е. присваивания параметрам сети некоторых малых случайных значений.

Обучение представляет, собой итерационную процедуру, которая при реализации на персональных компьютерах требует значительного времени. Скорость сходимости алгоритма обучения является одной из самых важных характеристик программ для моделирования нейронных сетей.

Тестирование обученной нейронной сети [39]

Для проверки правильности обучения построенной нейронной сети в нейроимитаторах предусмотрены специальные средства ее тестирования. В сеть вводится некоторый сигнал, который, как правило, не совпадает ни с

одним из входных сигналов примеров обучающей выборки. Далее анализируется получившийся выходной сигнал сети.

Тестирование обученной сети может проводиться либо на одиночных входных сигналах, либо на тестовой выборке, которая имеет структуру, аналогичную обучающей выборке, и также состоит из пар (<вход>, <требуемый выход>). Обычно, обучающая и тестовая выборки не пересекаются, та как тестовая выборка строится индивидуально для каждой решаемой задачи.

3.4. Исследование эффективности применения нейронной сети для обработки изображений на основе нейроиммитатора Сигнейро

В настоящее время известно большое количество нейропакетов, выпускаемых рядом фирм и отдельными исследователями и позволяющих конструировать, обучать и использовать нейронные сети для решения практических задач [39]. Однако, для мультимедийных приложений в свободном доступе был найден нейроиметатор Сигнейро. Поэтому рассмотрим его более подробно.

Сигнейро - нейроимитатор, позволяющий обучить нейронную сеть обработке изображений. Он предназначен для быстрого конструирования алгоритмов преобразования изображений. От пользователя требуется предоставить программе исходные изображения и результаты, которые должны получиться после их обработки. Обучение происходит на основе формируемых локальных признаков изображения, вычисляемых для каждой точки. Для обучения нейронной сети используется генетический алгоритм. После обучения автоматически создается динамическая библиотека с функцией преобразования (либо сегментации) изображения. Данную библиотеку с готовым алгоритмом можно подключить к собственным разработкам [40].

Дополнительно программа включает в себя два вспомогательных инструмента: “Обработка изображений” с основными функциями обработки изображений и “Разметка изображений” для формирования обучающей выборки. Программа окажет большую помощь при разработке программ преобразования различного класса изображений.

Рассмотрим работу нейронной сети на основе восстановления цвета полутоновых изображений [40]:.

Постановка задачи [40]**.** Для каждой точки полутонового изображения необходимо как можно более точно оценить значения красной, зеленой и синей составляющих, которые присутствовали на исходном цветном изображении, преобразованном к полутоновому. Далее этот процесс будем называть раскраской или раскрашиванием.

Метод восстановления цвета [40]**.** Основная идея раскраски, примененная автором, состоит в следующем: на цветных изображениях с однородным по типу содержанием (например, закат солнца, дорожное асфальтовое полотно и другие) однотипные объекты имеют примерно одинаковый цвет. Например, в летнем лесу все деревья зеленые; в осеннем - золотистого цвета; в зимнем - деревья покрыты снегом белого цвета; на голубом небе облака белые, а тучи серые; и так далее. Для оценки цвета каждой точки исходного полутонового изображения используется следующая информация относительно ее и самого изображения [40]:

1. Тип сцены изображения (определяет, что изображено на сцене).

2. Яркость точки.

3. Набор значений локальных признаков, которые позволят более точно определить вид объекта, и, соответственно, более точно “подобрать” цвет.

Тип сцены изображения предопределяет, какие объекты могут присутствовать на сцене. Яркость точки берется из входного полутонового изображения; значения локальных признаков вычисляются детерминированными алгоритмами обработки изображений. Непосредственно “подбор” цвета выполняет нейронная сеть, на вход которой

поступает набор значений локальных признаков и яркость точки, а на выходе формируются интенсивности цветовых компонент (красная, зеленая и синяя). Сеть предварительно обучается для каждого типа сцены.

Чтобы реализовать алгоритм раскраски изображения с заданным типом сцены выполняются три этапа [40]:

1. Формируется обучающая выборка.

2. Выполняется машинное обучение.

3. Результаты обучения в виде алгоритма прогона нейронной сети внедряются в программу.

В данной случае обучающей выборкой являются цветные изображения, их полутоновые копии и изображения локальных признаков, построенные для каждого полутонового изображения. Схематически формирование обучающей выборки отражено на схеме рис. 3.5.

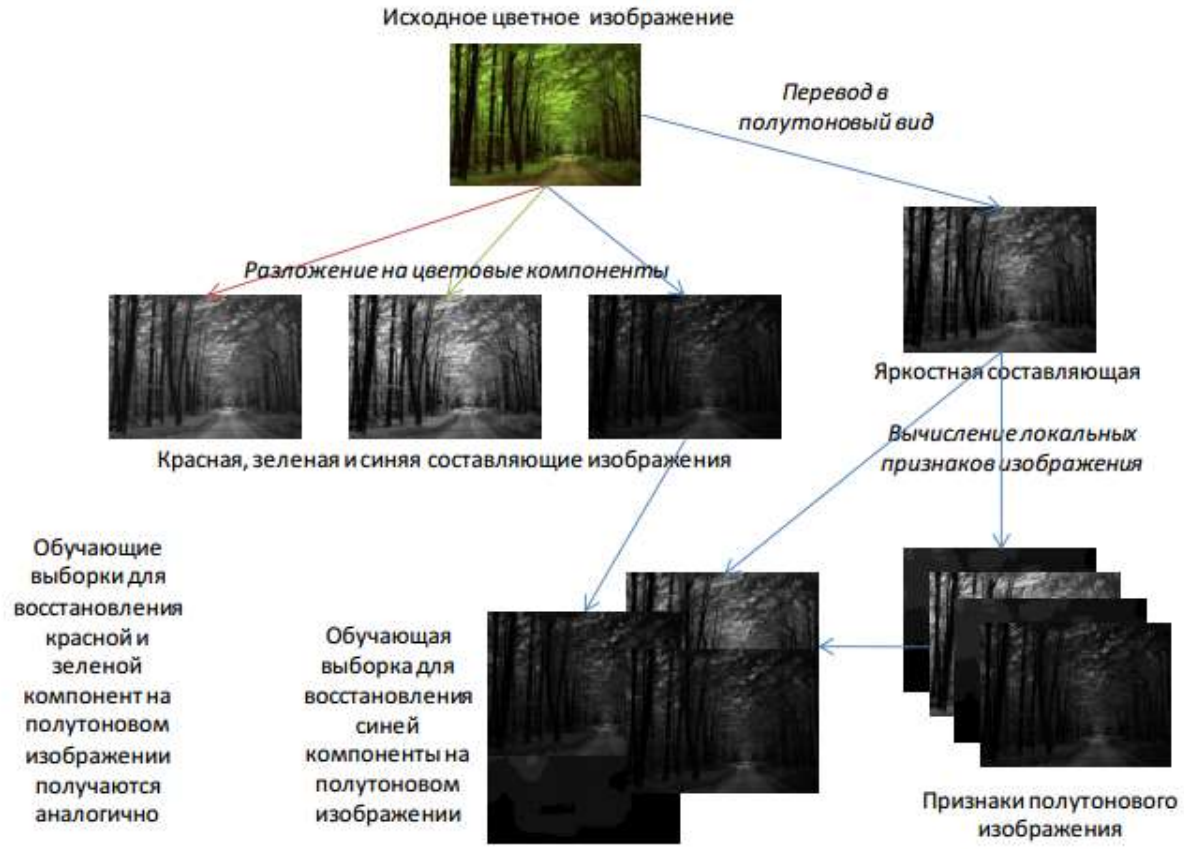


Рис. 3.5. Схема получения обучающей выборки для восстановления цветовых составляющих изображения

Далее нейронная сеть обучается восстанавливать каждую цветовую компоненту. Фактически для каждого типа сцены изображения обучаются три сети. Для реализации раскрашивания, к примеру, 10 типов сцен, необходимо обучить и программно сымитировать работу 30 нейронных сетей. Объединением результатов машинного обучения строится алгоритм раскрашивания по схеме рис. 3.6 [40].

Рис. 3.6. Схема получения алгоритма восстановления цвета полутонового изображения на основе результатов обучения нейронных сетей

После обучения можно практически применить алгоритм раскрашивания, который состоит из следующих шагов [40]:

1. Оператором выбирается тип сцены в соответствии с содержанием исходного полутонового изображения.

2. Строится набор изображений локальных признаков. Локальный признак 1 Локальный признак N … Яркость точки Интенсивность цветовой составляющей

3. Формируется красная компонента выходного цветного изображения путем прогона нейронной сети для каждой точки изображения.

4. Аналогично пункту 3 формируются зеленая и синяя компоненты выходного изображения.

5. Из композиции трех полученных составляющих собирается цветное изображение.

Для обучения сети использовался нейроимитатор Сигнейро 1.3, специально предназначенный для создания и обучения нейронных сетей обработке изображений. В результате применения подхода была создана экспериментальная программа восстановления цветности (раскрашивания) полутоновых изображений. В качестве базовых были выбраны десять типов сцен изображений (“Летний лес”, “Волны”, “Архивный текстовый документ” и другие). Результаты раскраски подобных полутоновых изображений приведены на рис. 3.7 [40].

А) Б) В)

Рис. 3.7. Восстановление цвета изображений природных объектов, где а) - исходные полутоновые изображения; б) - результаты восстановления цвета на изображениях рисунка (а) с выбором типов раскраски “волны” и “летний лес”; в) - оригинальные цветные изображения (эталоны для восстановления цвета изображений)

Для сравнения и оценки качества работы показаны и исходные цветные изображения. Можно видеть, что результаты близки к оригиналам. Но, не смотря на это, цветовая гамма на некоторых результирующих изображениях

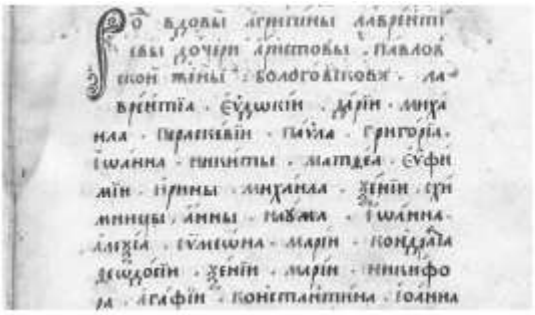
становится менее насыщенной. Получение более качественных результатов возможно после усложнения архитектуры нейронной сети и увеличению числа используемых признаков, но это приведет к повышению требований к ресурсам вычислительной системы. На данном этапе в практических целях можно использовать дополнительную цветокоррекцию изображения и “подкраску” отдельных деталей, проводимую вручную оператором.

Восстановление цвета в задачах реставрации архивных текстовых документов [40]**.** В настоящее время многие музеи используют электронные коллекции исторических документов, пополняемые путем оцифровки исходных документов. Это позволяет обеспечить непосредственный доступ к ним широкой публики без (нежелательного) использования оригинальных документов (создание иллюстрированных каталогов на CD/DVD, размещение изображений в сети Интернет). Однако, проблемой является плохое качество многих исторических документов, в том числе уникальных. Как правило, цифровую реставрацию проводят с использованием стандартных программных средств обработки изображений (например, Adobe Photoshop). Но такие средства не используют алгоритмы восстановления цветов. Кроме того, организация машинного обучения на целых коллекциях им не свойственна и не предвидится в ближайшем будущем. Один их лучших результатов применения разработанного метода был получен на изображениях архивных текстовых документов (рис. 3.8).

Как видно по рис.3.8, метод позволяет не только восстановить цвет документа, но и перекрасить его, придав ему другой облик.

Ночное видение в цвете [40]**.** Изображения с цифровых приборов ночного видения характеризуются узким яркостным диапазоном и отсутствием цветовой составляющей. Прибором регистрируется только яркость точек изображения как результат свечения ускоренных фотоэлектронов на люминесцентном экране. Соответственно, на выходе наблюдается либо полутоновое изображение, либо изображение в видимой

глазу области спектра (чаще всего зеленой). Создать условия для лучшего восприятия этого изображения человеком можно, если цвета объектов сцены сделать привычными для человека. Ввиду отсутствия возможности создания алгоритмов корректного раскрашивания произвольной сцены, ее тип задается пользователем. Поэтому предложенный метод удобно применять для работы с приборами, статично установленными для ночного наблюдения за сценой. В данной задаче разработанный метод раскрашивания целесообразно применять в сочетании с методами видоизменения гистограмм яркости, так как исходные изображения характеризуются узким диапазоном яркостей.



а) Исходное полутоновое изображение,

б) Восстановление цвета на изображении (а) (вариант 1)

в) Восстановление цвета на изображении (а) (вариант 2),

г) Восстановление цвета на изображении (а) (вариант 3)

Рис. 3.8. Восстановление цвета изображения архивного текстового документа

Применение метода раскрашивания к изображениям, полученным с приборов ночного видения, демонстрируется на рис. 3.9.

а) Исходное полутоновое изображение,

б) Восстановление цвета на изображении (а) (вариант 1)

в) Восстановление цвета на изображении (а) (вариант 2),

г) Восстановление цвета на изображении (а) (вариант 3)

Рис. 3.9. Восстановление цвета изображения архивного текстового документа

В настоящее время доступно две версии данной программы:

1) Сигнейро 1.2 Професиональная – платная версия для профессиональной работы и создания коммерческих приложений на базе программы.

2) Сигнейро 1.2 Учебная – академическая версия для работы в учебных заведениях. Это бесплатный вариант с ограниченными функциями. Её рекомендуется использовать в высших учебных заведениях на технических специальностях при изучении предметов, связанных с обработкой изображений и искусственным интеллектом.

Выводы

На основе проведённого анализа принципов работы нейронных сетей, установлено, что:

- Для их функционирования требуется очень большое число сложных математических вычислений, многие из которых необходимо выполнять параллельно.
- Для повышения скорости вычислений разрабатываются и активно используются специальные нейропроцессоры и нейрокомпьютеры позволяющие достаточно эффективно решать сложные задачи по распознаванию изображений и звука.
- Для оценки эффективности использования нейросетей для сжатия объемов данных изображений был проведен анализ экспериментальных данных на примере нейроиммитатора Сигнейро, в котором реализуются методы машинного обучения следующих задач [41]:
  - Обработка изображений (в том числе нелинейная фильтрация и сегментация).
  - Классификация изображений.
  - Колоризация (восстановление цвета) полутоновых изображений.
  - Определение тематической близости между двумя изображениями.
- Результаты работы Сигнейро применимы в следующих областях [41]:
  - Реставрация изображений фотографических и текстовых документов;
  - Раскрашивание полутоновых изображений;
  - Дефектометрия (сегментация дефектных участков для их дальнейшего анализа);
  - Распознавание текста и подготовка к распознаванию текстовых документов;

- Разработка систем технического зрения;
- Индексация изображений в информационно-поисковых системах;
- Поиск спама в изображениях;
- Классификация и кластеризация изображений;
- Автоматическая обработка медицинских снимков;
- Улучшение качества изображений;
- Решение многих других задач, связанных с обработкой и анализом изображений.

4. ПРИМЕНЕНИЕ ИСКУСТВЕННЫХ НЕЙРОСЕТЕЙ ДЛЯ СЖАТИЯ ВИДЕОДАННЫХ ТВ ИЗОБРАЖЕНИЙ И АНАЛИЗ ИХ ЭФФЕКТИВНОСТИ

4.1. Применение нейронной сети Кохонена для сжатия видеоданных изображений и оценка эффективности ее работы

В настоящее время развитие искусственных нейронных сетей достигло такого уровня, что они уже начинают с успехом применятся для различного рода обработки изображений с целью реконструкции, улучшения их качества или распознавания видеобъектов. Однако, применение их для сжатия объемов видеоданных ТВ изображений пока ограниченно. Это связано с тем, что на сегодняшний день разработанные нейрочипы имеют низкий уровень интеграции, а нейро имитаторы имеют низкое быстродействие. Кроме того, как показывает практика, много неточностей связано с оценкой возможности ИНС сжимать информацию. При подсчете коэффициента компрессии зачастую не учитываются реальные затраты на хранение и передачу весовых коэффициентов обученной ИНС, что дает искаженное представление о реальной степени сжатия. Тем не менее работы в этом направлении исследований ведутся, о чем свидетельствуют работы по применению сети Кохонена для сжатия объемов данных изображений [43]. Рассмотрим эффективность применения сети для сжатия изображений на основе векторного квантования. Данная технология основана на способе кластеризации, в котором пространство входов делится на ряд областей, для каждой из которых определяется вектор восстановления. Архитектура самоорганизующейся карты признаков для векторного квантования определяется размером словаря кодовых векторов. Каждый кодовый вектор - это матрица весов в соревновательном слое. В эксперименте блок, содержащий 16 пикселей, подавался на слой Кохонена, состоящий из 256 узлов-кластеров, размещенных в двумерном массиве 16x16. Веса,

связывающие j-й нейрон слоя Кохонена и входы, представлены матрицей $[w_{ij}], j = 0,1,...,255, i = 0,1,...,15$.

Алгоритм сжатия изображения использует следующие действия[43]:

1) изображение делится на блоки, подаваемые в случайном порядке на вход сети;

2) выбирается нейрон с минимальным евклидовым расстоянием (геометрическое расстояние между двумя точками в многомерном пространстве, вычисляемое по теореме Пифагора) до поданного блока;

3) веса нейронов, не победивших хотя бы у одного блока, исключаются из словаря;

4) сохраняется множество индексов соответствия нейронов-победителей блокам сжатого изображения;

5) сохраняется словарь кодовых векторов.

Алгоритм восстановления изображения:

1) для каждого индекса блока сжатого изображения находим соответствующий кодовый вектор нейрона-победителя;

2) найденный кодовый вектор формирует блок результирующего изображения.

Для сжатия использовались файлы изображений размером 2112x2816 пикселей, то есть объемом 5947392 пикселей. Коэффициент сжатия рассчитывается по формуле [43]:

$$G = \frac{O}{NS\frac{|log_2 N|}{8}B} \quad (4.1)$$

где O - площадь сжимаемого изображения, N - размер словаря, S - длина кодовых векторов (площадь блока), B - число блоков в изображении. Результаты эксперимента при N=25 представлены на рис 4.1 [43].

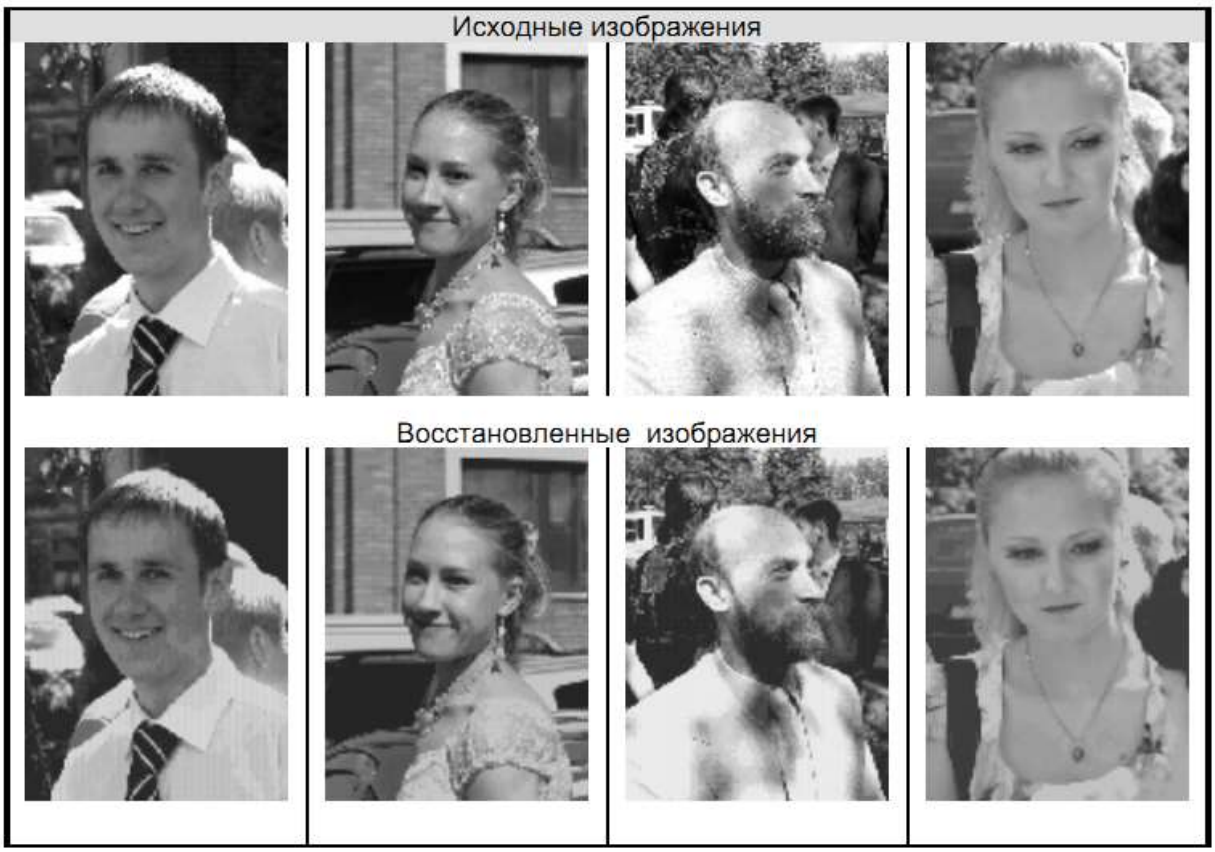


Рис 4.1. Результаты эксперимента при размере словаря кодовых векторов равном N=25

Эксперименты показывают, что восстановленные изображения обладают приемлемым качеством даже при высоких степенях сжатия [43].

На рис.4.2 приведены результаты сравнения качества сжатия изображений по Кохонену и широко известного алгоритма JPEG-2000 на основе оценки среднеквадратического отклонения (СКВО) значений пикселей при различных значениях коэффициента сжатия G [43]. Для визуальной оценки качества сжатия с помощью ИНС Кохонена на рис. 4.3 показаны фрагменты оригинального и восстановленного изображений для G=33. При сильном увеличении фрагмента восстановленного изображения хорошо заметны последствия ошибок сети – «лестничный» эффект, вызванный визуализацией границ непересекающихся блоков. Эксперимент показывает, что восстановленные изображения обладают приемлемым качеством.

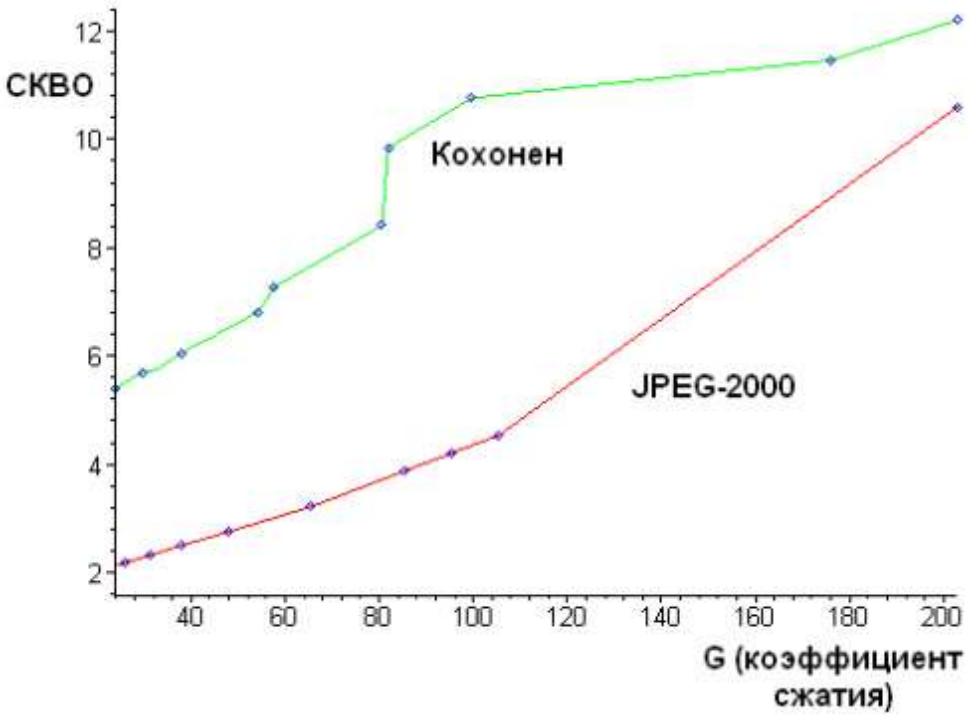


Рис.4.2. Сравнение алгоритмов сжатия по качеству восстановленных изображений

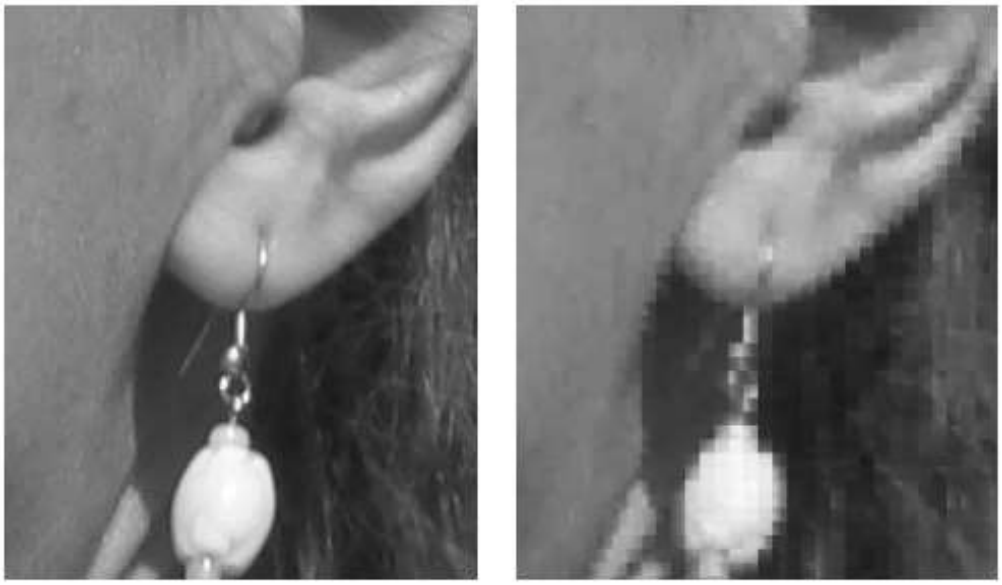


Рис.4.3. Сравнительное качество исходного и восстановленного изображения

4.2. Анализ эффективности сжатия изображений рециркуляционными нейронными сетями

Рециркуляционные нейронные сети представляют собой многослойные нейронные сети с обратным распространение информации [44]. При этом обратное распространение информации происходит по двунаправленным связям, которые имеют в различных направлениях разные весовые коэффициенты. При обратном распространении сигналов, в таких сетях осуществляется преобразование их с целью восстановления входного образа. В случае прямого распространения сигналов происходит сжатие входных данных. Обучение рециркуляционных сетей производится без учителя.

Рециркуляционные сети характеризуются как прямым Y=f(X), так и обратным X=f(Y) преобразованием информации. Задачей такого преобразования является достижение наилучшего автопрогноза или самовоспроизводимости вектора X. Рециркуляционные нейронные сети применяются для сжатия (прямое преобразование) и восстановления исходной (обратное преобразование) информации. Такие сети являются самоорганизующимися в процессе работы. Они были предложены в 1988 году. Теоретической основой рециркуляционных нейронных сетей является анализ главных компонент [44].

Метод главных компонент применяется в статистике для сжатия информации без существенных потерь ее информативности. Он состоит в линейном ортогональном преобразовании входного вектора X размерности n в выходной вектор Y размерности p, где $p<n$. При этом компоненты вектора Y являются некоррелированными и общая дисперсия после преобразования остается неизменной. Совокупность входных паттернов представим в виде матрицы [44]:

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{L1} & x_{L2} & \dots & x_{Ln} \end{bmatrix} \quad (4.2)$$

Где $x^k = (x_{k1}, x_{k2}, \dots, x_{kn})$ соответствует k-му входному образу, L — общее количество образов.

Будем считать, что матрица X является центрированной, то есть вектор математических ожиданий μ=0. Этого добиваются при помощи следующих преобразований [44]:

$$x_{ij} = x_{ij} - \mu_j$$

$$\mu_j = \sum_{i=1}^{L} \frac{x_{ij}}{L} \quad (4.3)$$

Матрица ковариаций входных данных X определяется как [44]:

$$K = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \dots & \sigma_{2n} \\ \dots & \dots & \dots & \dots \\ \sigma_{n1} & \sigma_{n2} & \dots & \sigma_{nn} \end{bmatrix} \quad (4.4)$$

где σij — ковариация между i-ой и j-ой компонентой входных образов.

Элементы матрицы ковариаций можно вычислить следующим образом:

$$\sigma_{ij} = \frac{1}{L} \sum_{k=1}^{L} (x_i^k - \mu^i) * (x_j^k - \mu^j) \quad (4.5)$$

где i,j = 1,…,n.

Метод главных компонент состоит в нахождении таких линейных комбинаций исходных переменных [44]:

$$\begin{aligned} y_1 &= w_{11}x_1 + w_{21}x_2 + \dots + w_{n1}x_n \\ y_2 &= w_{12}x_1 + w_{22}x_2 + \dots + w_{n2}x_n \\ y_p &= w_{1p}x_1 + w_{2p}x_2 + \dots + w_{np}x_n \end{aligned} \quad (4.6)$$

Что

$$\sigma(y_i, y_j) = 0; i, j = \overline{1,n}$$
$$\sigma(y_1) \geq \sigma(y_2) \geq \cdots \geq \sigma(y_p)$$
$$\sum_{i}^{n} \sigma_{ii} = \sum_{i}^{n} \sigma(y_i) \quad (4.7)$$

Из последних выражений следует, что переменные yi некоррелированы, упорядочены по возрастанию дисперсии и сумма дисперсий входных образов остается без изменений. Тогда подмножество первых p переменных y характеризует большую часть общей дисперсии. В результате получается представление входной информации.

Переменные y, i = 1,…,p называются главными компонентами. В матричной форме преобразование главных компонент можно представить как [44]:

$$Y = F(W^T X) \quad (4.8)$$

где строки матрицы WT должны удовлетворять условию ортогональности, т.е:

$$W_i W_j^T = 1, \forall i = j$$
$$W_i W_j^T = 0, \forall i \neq j \quad (4.9)$$

при этом вектор Wi определяется как:

$$W_i = (w_1, w_2, \ldots, w_{ni}) \quad (4.10)$$

Для определения главных компонент необходимо определить весовые коэффициенты Wi,j = 1,…,p.

Рециркуляционная нейронная сеть представляет собой совокупность двух слоев нейронных элементов, которые соединены между собой двунаправленными связями.

Каждый из слоев нейронных элементов может использоваться в качестве входного или выходного. Если слой нейронных элементов служит в качестве входного, то он выполняет распределительные функции.

В противном случае нейронные элементы слоя являются обрабатывающими. Весовые коэффициенты соответствующие прямым и

обратным связям характеризуются матрицей весовых коэффициентов W и W’. Для наглядности, рециркуляционную сеть можно представить в развернутом виде (Рис 4.4) [44].

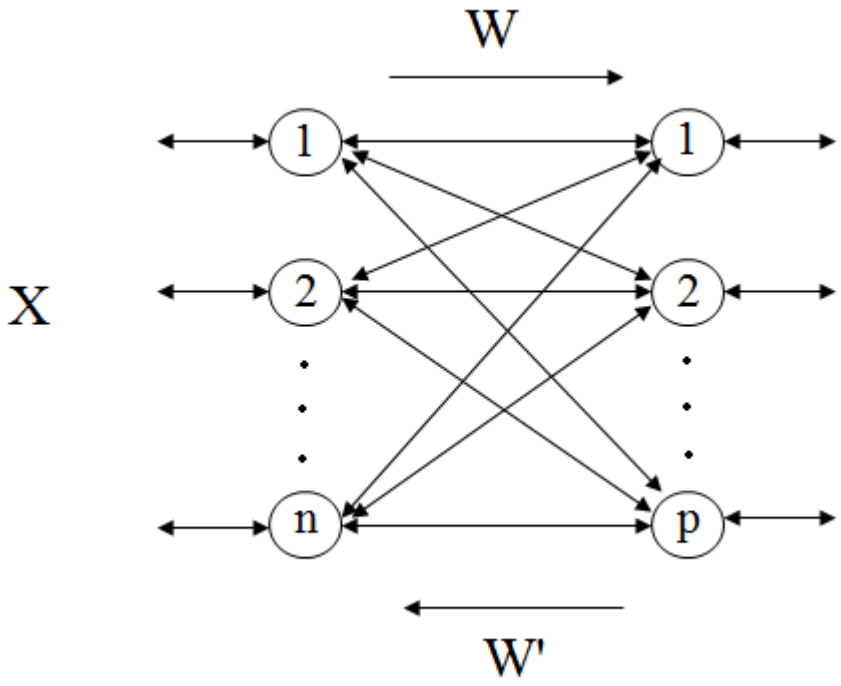
Такое представление сети является эквивалентным и характеризует полный цикл преобразования информации. При этом промежуточный слой нейронных элементов производит кодирование (сжатие) входных данных X, а последний слой осуществляет восстановление сжатой информации Y. Назовем слой нейронной сети, соответствующий матрице связи W прямым, а соответствующий матрице связей W’ — обратным [44].

Рециркуляционная сеть предназначена как для сжатия данных, так и для восстановления сжатой информации. Сжатие данных осуществляется при прямом преобразовании информации в соответствие с выражением [44]:

$$Y = F(W^T X) \tag{4.11}$$

Восстановление или реконструкция данных происходит при обратном преобразовании информации:

$$X = F(W' X) \tag{4.12}$$



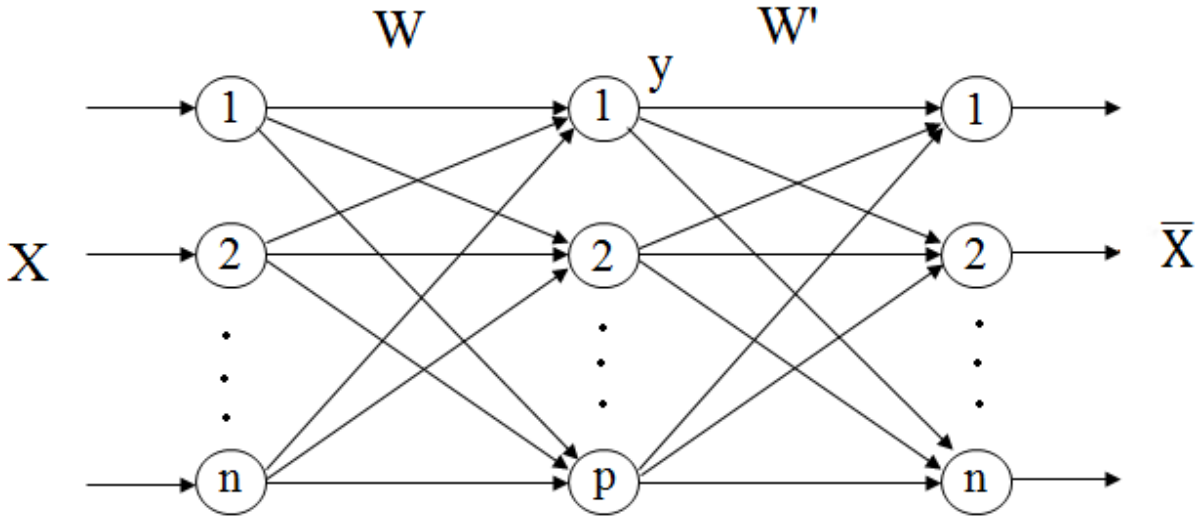


Рис. 4.4. Пример рециркуляционной нейронной сети

В качестве функции активации нейронных элементов F может использоваться как линейная, так и нелинейная функции. При использовании линейной функции активации:

$$Y = W^T X$$
$$\bar{X} = W' X \qquad (4.13)$$

Линейные рециркуляционные сети, в которых весовые коэффициенты определяются в соответствии с методом главных компонент называются PCA сетями.

Рециркуляционные нейронные сети можно применять для сжатия и восстановления изображений. Обработка изображений начинается с разделением его на блоки, при этом блоком называется окном, которому в соответствие ставится рециркуляционная нейронная сеть. Количество нейронов первого слоя сети соответствует размерности окна (количеству пикселей; в некоторых случаях каждый цвет обрабатывается отдельно). Сканируя изображение при помощи окна и подавая значения его пикселей на нейронную сеть, можно сжать входное изображение. Сжатое изображение можно восстановить при помощи обратного распространения информации [44].

А) исходное изображение Б) восстановленное изображение

Рис 4.5. Пример обработки изображений рециркуляционной нейронной сети

Для приведенного примера (рис 4.5) использовалось окно размером 3 на 3 пикселя, количество нейронов на втором слое — 21, максимальная допустимая ошибка — 50. Коэффициент сжатия составил 1,3 роза. Для обучения нейронной сети потребовалось 129 повторений [44].

4.3. Рекомендации по использованию нейронных сетей в цифровом телевидении

Несколько десятилетий назад было положено начало исследованиям методов обработки информации, называемых сегодня нейросетевыми. С течением времени интерес к нейросетевым технологиям то ослабевал, то вновь возрождался. Такое непостоянство напрямую связано с практическими результатами проводимых исследований [26].

Сегодня исследования в области искусственных нейронных сетей (ИНС) обрели заметную динамику. Подтверждением тому служит факт финансирования этих работ в США, Японии и Европе, объем которого

исчисляется сотнями миллионов долларов [26]. Растет число публикаций по тематике ИНС, широк и их спектр: от монографий и статей, единодушно признанных основополагающими в данной области [26], до обзоров, посвященных прикладным вопросам [26]. Издается несколько журналов, посвященных тематике ИНС, таких, например, как "Приборостроение" и "Нейрокомпьютер", IEEE Transaction on NeuralNetworks, Neural Networks, Neural Computing & Applications.

Вместе с тем реальные результаты практического применения нейросетевых технологий, пока немногочисленны. Отчасти это объясняется следующими причинами [26]:

- использование аппарата ИНС имеет свои особенности, которые несвойственны традиционным методам;
- путь от теории нейронных сетей к их практическому использованию требует соответствующей адаптации методологий, отработанных первоначально на модельных задачах;
- вычислительная техника с традиционной архитектурой не лучшим образом приспособлена для реализации нейросетевых методов.

При этом можно выделить ряд задач, которые могут решаться с использованием искусственных нейронных сети:

- **Задачи обработки видеоизображений.** Одной из наиболее сложных и актуальных задач обработки видеоизображений, представленных последовательностью оцифрованных кадров, является проблема выделения и распознавания движущихся объектов в условиях действия различного рода помех и возмущений. Для ее решения разработана специализированная система, которая осуществляет выделение изображений движущихся объектов на сложном зашумленном фоне, фильтрацию помех, скоростную фильтрацию, отделение объектов от фона, оценку скорости каждого объекта, его идентификацию и сопровождение. Система построена с применением нейросетевых методов и работает с реальными данными телевизионной системы (25 кадров/с, 320x200 пикселов) [26].

Выделение изображений движущихся объектов осуществляется путем построения оценки поля скоростей с помощью многослойной локально-связной нейронной сети оригинальной конструкции. Размерность сети для изображения 320x200 пикселов составляет несколько миллионов нейронов и примерно вчетверо больше синапсов [26].

Распознавание выделенных силуэтов производится на самоорганизующейся нейронной сети, предварительно обученной на изображениях объектов рассматриваемых классов. Система инвариантна к произвольному движению фона, зашумлению белым шумом до 10%. Вероятность правильного распознавания составляет около 90% [26].

Система реализована на обыкновенном ПК и специально разработанном программно-аппаратном комплексе, обеспечивающем обработку информации в реальном времени.

- **Задачи обработки статических изображений.** Не менее сложными являются задачи выделения и распознавания объектов на статическом тоновом изображении. В частности, подобные задачи возникают при автоматической обработке спутниковых изображений земной поверхности. Для их решения разработана и реализована на ПК автоматизированная система анализа изображений земной поверхности, полученных в оптическом диапазоне с искусственного спутника Земли. Система в автоматическом режиме обеспечивает выделение на обрабатываемых изображениях объектов заданных классов: дорожной сети, кварталов с характерной застройкой, аэродромов и стоящих на них самолетов [26].

Нейросетевые принципы, заложенные в систему, позволяют проводить ее обучение и переобучение. Система инвариантна к яркостным характеристикам объектов.

- **Многопроцессорные ускорительные платы.** Одной из особенностей нейросетевых методов обработки информации является высокая параллельность вычислений и, следовательно, целесообразность

использования специальных средств аппаратной поддержки [26]. В значительной мере успех в решении рассмотренных задач обусловлен использованием оригинальных ускорительных плат. Такие платы работают параллельно с процессором обыкновенного ПК и несут на себе основную вычислительную нагрузку, превращая основной процессор компьютера в устройство управления и обслуживания мощных вычислительных средств, расположенных на ускорительной плате.

В НТЦ "Модуль" разработаны многопроцессорные ускорительные платы МЦ5.001 и МЦ5.002 [26]. Первая из них имеет в своем составе 4 микропроцессора TMS320C40 с тактовой частотой 50 МГц и пиковой производительностью 275 MIPS. Каждый процессор имеет свою локальную статическую память объемом 1 Мбайт. К 2 процессорам дополнительно подключены 2 блока динамической памяти объемом 16 Мбайт каждый. К одному из процессоров подключена также статическая память объемом 1 Мбайт, используемая для обмена данными с ПК. Процессоры соединены друг с другом специальными высокоскоростными каналами с пропускной способностью 20 Мбайт/с каждый. Наращивание и комплексирование плат осуществляется на материнской плате ПК с помощью шины ISA.

Ускорительная плата МЦ5.002 содержит 6 процессоров TMS320C40 и выполнена в конструктиве VME, что позволяет использовать ее в бортовых системах, расположенных на летательном аппарате [26].

- **Нейропроцессор.** Ускорительные платы МЦ5.001 и МЦ5.002 повышают эффективность использования нейросетевых методов обработки информации. Однако существующая тенденция к возрастанию объемов вычислений приводит к необходимости дальнейшего наращивания производительности нейровычислителей. В связи с этим в НТЦ "Модуль" разработан собственный нейропроцессор [26], совмещающий в себе как универсальный вычислитель, так и специализированное вычислительное устройство, ориентированное на выполнение базовых нейросетевых операций.

Нейропроцессор состоит из двух основных блоков: скалярного, выполняющего роль универсального вычислительного устройства, и векторного, ориентированного на выполнение векторно-матричных операций. Скалярное устройство обеспечивает интерфейсы с памятью и 2 коммуникационными портами, позволяющими объединять процессоры в вычислительные сети различной конфигурации. Основное назначение скалярного устройства - подготовка данных для векторной части процессора. Для этого существует несколько режимов адресации, интерфейс с памятью, наборы арифметических и логических операций, возможность работы с регистровыми парами [26]. Скалярное устройство имеет адресных регистров и такое же количество регистров общего назначения разрядностью 32 бита каждый.

Центральным звеном нейропроцессора является целочисленное векторное устройство, обладающее возможностями обработки данных различной разрядности. Оно оперирует 64-разрядными словами, которые могут быть разбиты на целочисленные составляющие практически произвольной разрядности в пределах от 1 до 64 бит. На каждую инструкцию векторного процессора затрачивается от 1 до 32 тактов [26]. При этом одновременно обрабатывается до 32 64-разрядных слов. Для организации непрерывной подачи данных в операционное устройство (ОУ) векторного процессора используются внутренние блоки памяти, называемые векторными регистрами. Они выполняют роль буфера операндов, буфера для хранения матрицы весов, очереди результатов. При выполнении команды в операционном устройстве операнды по очереди извлекаются из внутреннего буфера и подаются на один из входов ОУ. Внутри ОУ производятся вычисления, а их результат заносится в буфер результатов. Векторные инструкции, хотя и занимают несколько тактов процессорного времени, могут выполняться параллельно с инструкциями скалярного процессора. Таким образом, процессор рассчитан на высокопроизводительную обработку больших массивов целочисленных данных [26].

Нейропроцессор выполнен по технологии 0,5 мкм. Его тактовая частота 33 МГц. На специальных векторно-матричных операциях он дает увеличение производительности в десятки раз по сравнению с процессором TMS320C40. Благодаря наличию коммуникационных портов с интерфейсом, идентичным портам TMS320C40, нейропроцессор может быть интегрирован в гетерогенную многопроцессорную систему. Примеры построения подобных систем приводятся в работе [26].

Для нейропроцессора разработан полный пакет системного программного обеспечения, включая символьный отладчик, и ряд прикладных библиотек, в частности библиотеку векторно-матричных вычислений.

- **Создание программных средств аппаратной поддержки нейровычислений.** Специфика рассматриваемых вычислительных средств и решаемых задач обуславливает новые требования к технике программирования. Программисту приходится оперировать другими категориями, по-другому строить логику программы, решать задачи, которые не могли возникнуть при традиционном программировании [26]. Перед ним стоит задача - максимально эффективно использовать ресурсы вычислительной системы, правильно распределить нагрузку между процессорами, задействовать их специфичные возможности**.**

Здесь на первый план выходят методы параллельной обработки данных ориентированные, как обработку на параллельно работающих процессорах, так и одновременную обработку нескольких элементов данных на одном процессоре. Современный процессор позволяет выполнять несколько инструкций за один такт, что заставляет программиста продумывать как способы организации самих вычислений, так и способы подготовки данных, для того чтобы параллельно выполняемые процессы не блокировали друг друга.

Трудности, возникающие при программировании многопроцессорных систем, хорошо известны: синхронизация параллельных процессов,

механизмы обмена данными, проблемы, когда несколько процессов задействуют одни и те же ресурсы. Еще одной важной особенностью современных процессоров является высокая разрядность операндов, например 64 бита, что позволяет размещать в них по нескольку малоразрядных элементов данных и обрабатывать их параллельно.

Примером эффективного использования отмеченной особенности современных процессоров является технология MMX [26], где 64-разрядный регистр разбивается на 8 независимых байтов или на 4 16-битных слова, которые обрабатываются параллельно. Независимость элементов состоит в том, что при смещениях или вычитании не происходит заимствования битов у соседних элементов. В НТЦ "Модуль" пошли еще дальше - в кристалл заложена функция произвольного разбиения 64-разрядного слова на элементы разрядностью от 1 до 64, а также возможность располагать в одном длинном слове данные разной разрядности, разбивая слово произвольным образом. В результате программист в зависимости от разрядности исходных данных может варьировать количество параллельно обрабатываемых элементов.

Как показал анализ результатов эффективности применения нейронных сетей для сжатия изображений, проведенный по доступным литературным источникам, эффективность сжатия на сегодняшний день довольно низкая. Однако данное направление исследований находится в ранней стадии развития и в дальнейшем, с развитием нейропроцессоров и нейротехнологий можно ожидать существенного прорыва в эффективность обработки изображений и звука.

Таким образом, нейрокомпьютеры являются перспективным направлением в развитии новейших технологий. В настоящее время ведутся дальнейшие разработки в области нейронных сетей, которые направленны на улучшение качества видео и звука.

Выводы

Для оценки эффективности использования нейросетей для сжатия объемов данных изображений был проведен анализ экспериментальных данных в доступных литературных источниках. Анализ был проведен на примере двух нейронных сетей: Сети Кохонена и рециркулярных нейронных сетях. В результате рассмотренных экспериментов было установлено, что на данное время качество сжатия и восстановления изображений на исскуственных нейронных сетях уступают при прочих равных условиях известному алгоритму JPEG-2000. Однако при этом искусственные нейронные сети обеспечивают уникальность для установленной серии изображений набора весовых коэффициентов, что может быть использовано при шифровании передаваемой и восстанавливаемой информации [43].

Возможности распараллеливания задач с использованием искусственных нейронных сетей достаточно высоки, что обеспечивается аппаратно-программной поддержкой, таких как нейрокомпьютеры, нейпроцессоры и различного рода нейронные ускорители, обеспечивающих масштабируемое ускорение вычислений.

Приведенные результаты экпериментов не являются исчерпывающими в исследовании возможностей нейронных сетей, а также эффективности их применения. В настоящее время продолжается работа над совершенствованием модулей и алгоритмов решения прикладных задач на основе искусственных нейронных сетей [43]. Возможные области применения искусственных нейронных сетей: задачи телевизионного вещания, медицине, в задачах аэрокосмитического назначения: обработка информации, применение искусственных нейронных сетей в задачах обнаружения и распознавания локальных объектов, прогнозирование динамических объектов, а также решение траекторных задач в системах управления и т.д .

ЗАКЛЮЧЕНИЕ

Развитие телевизионной отрасли, как средства массовой информации и особенно телевизионных технологий развлекательных передач, требуют резкого увеличения потока видеоданных, что ведет к значительному увеличению объемов информации изображения и звука, которые необходимо передавать по существующим каналам связи. Поэтому для согласования параметров сигналов и каналов связи используется сжатие объемов данных.

Установлено, что телевизионные изображения обладают рядом типов избыточной информации: кодовая, межэлементная или статистическая, психовизуальная, структурная и временная или межкадровая, устранение которой позволяет существенно снизить объем передаваемых данных. Причем, поскольку телевидение является системой визуального наблюдения, то при сжатии объемов видеоданных учитываются особенности нашего зрительного восприятия, позволяющие исключить из изображения часть информации не воспринимаемую зрительной системой.

В настоящее время для сжатия изображений используются методы, основанные на спектральных преобразованиях, к которым относятся ДКП, вейвлет-преобразованиях и ряд других ортогональных функций, а также использующие фракталы и векторное квантование. Кроме, того, поскольку в телевидении информация в смежных кадрах обычно мало изменяется, то основное сжатие видеопотока обеспечивается за счет передачи только межкадровых различий.

Существующие методы сжатия видео не позволяют обеспечить высокое качество изображений и звукового сопровождения при больших коэффициентах сжатия видеопотока, что приводит к возникновению блочных искажений или потери четкости. Поэтому для повышения эффективности сжатия видео данных ведутся активные работы по разработке новых методов сжатия изображений, основанные на восприятии окружающего мира человеческим мозгом. Главным отличием нейронных сетей является её

обучаемость, которая состоит в корректировке весов связей, в результате которой каждое входное воздействие приводит к формированию соответствующего выходного сигнала.

Соответственно создание новых методов обработки данных предусматривает также создание необходимого оборудования, к которому относятся нейропроцессоры и нейрокомпьютеры.

Нейрокомпьютинг - это научное направление, занимающееся разработкой вычислительных систем шестого поколения - нейрокомпьютеров, которые состоят из большого числа параллельно работающих простых вычислительных элементов (нейронов) [35]. Большое число параллельно работающих вычислительных элементов обеспечивают высокое быстродействие.

Нейрокомпьютеры позволяют с высокой эффективностью решать целый ряд интеллектуальных задач, таких как: задачи распознавания образов, адаптивного управления, прогнозирования, диагностики и т.д.

В настоящее время для обработки видеоданных разработаны и активно применяются нейропроцессоры Л1879ВМ1 и Л1879ВМ3, российского производства НТЦ «Модуль». Отличительной особенностью процессора Л1879ВМ1 является его возможность программного управления делением 64-х разрядной сетки процессора, за счет векторного сопроцессора, что позволяет на одном процессоре одновременно реализовывать вычисления, соответствующие нескольким нейронам. В свою очередь микросхема Л1879ВМ3 является законченной системой на кристалле, предназначенной для обработки широкополосных квадратурных сигналов и синтеза аналоговых высокочастотных сигналов. Данная микросхема потребляет меньше энергии, обладает большей надежностью и помехозащищенностью, а также уменьшается время распространения сигнала, за счёт построения схемы на кристалле.

Для оценки эффективности использования нейросетей для сжатия объемов данных изображений был проведен анализ экспериментальных

данных в доступных литературных источниках. Анализ был проведен на примере нейроиммитатора Сигнейро, в котором реализуются методы машинного обучения следующих задач [41]:

- Обработка изображений (в том числе нелинейная фильтрация и сегментация).
- Классификация изображений (определение принадлежности изображения к определенному классу).
- Колоризация (восстановление цвета) полутоновых изображений.
- Определение тематической близости между двумя изображениями.

Как показал анализ результатов эффективности применения нейронных сетей для сжатия изображений, проведенный по доступным литературным источникам, эффективность сжатия на сегодняшний день довольно низкая. В результате рассмотренных экспериментов было установлено, что на данное время качество сжатия и восстановления изображений на исскуственных нейронных сетях уступают при прочих равных условиях известному алгоритму JPEG-2000. Однако при этом искусственные нейронные сети обеспечивают уникальность для установленной серии изображений набора весовых коэффициентов, что может быть использовано при шифровании передаваемой и восстанавливаемой информации [43]. Однако данное направление исследований находится в ранней стадии развития и в дальнейшем, с развитием нейропроцессоров и нейротехнологий можно ожидать существенного прорыва в эффективность обработки изображений и звука.

Таким образом, нейрокомпьютеры являются перспективным направлением в развитии новейших технологий. В настоящее время ведутся дальнейшие разработки в области нейронных сетей, которые направленны на улучшение качества видео и звука.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ:

1. Доклад Президента Республики Узбекистан Ислама Каримова на заседании Кабинета Министров, посвященном итогам социально-экономического развития страны в 2014 году и важнейшим приоритетным направлениям экономической программы на 2015 год. Национальное информационное агенство Узбекистана http://uza.uz/ru/politics/-respubliki-uzbekistan-islama-karimova-na-z-17-01-2015
2. А.Ю. Тропченко, А.А. Тропченко. Учебное пособие по дисциплине «Теоретическая информатика» «Методы сжатия изображений, аудиосигналов и видео». Санкт-Петербург 2009г.
3. Нгуен Виет Хунг. Автореферат Диссертации «Нейросетевые алгоритмы для решения задач кодирования изображений с использованием технологии CUDA». Москва – 2012
4. Лекция 9 "Сжатие данных" http://www.victoria.lviv.ua/html/informatika/lecture9.htm
5. И.А. Гаврилов И.А., Т.Г.Рахимов. Тезис доклада «Проблемы большого сжатия ТВ изображений» в сборнике Республиканской научно-технической конференции, проходившей 14марта 2013г., Том 4 стр.180-182,2013г.
6. В. М. Артюшенко, О. И. Шелухин, М. Ю. Афонин Учебное пособие «Цифровое сжатие видеоинформации и звука» И.: Москва 2003г. с 430
7. Гаврилов И.А., Ибраева С.М., Игнатьева О.С. «Особенности передачи ТВ сигналов по каналам сотовой связи».// Труды международной научной конференции «Роль и значение телекоммуникаций и информационных технологий в современном обществе» Ташкент 2006. Том-1 с 138.
8. Савицкая Д.А., В.Г. Маркинг. Анализ методов обработки ТВ изображений в стандарте MPEG-4. Тезисы в сборнике «Ахборот –

коммуникация технологиялари» аспирант, магистрант ва иктидорли талабаларининг илмий – техник конференцияси. Ташкент, 9-10 апрель 2009 г. Стр.77-80

9. Компьютерный анализ изображений: общие сведения, системы, примеры использования http://www.infectology.ru/ microscopy/ today/analysis/read_analysis7.aspx
10. Семенов Ю.А. Стандарт MPEG-4. http://book.itep.ru/2/25/mpeg-4R.htm#20
11. С.Уэлстид «Фракталы и вейвлеты для сжатия изображений в действии», ООО «Издательство ТРИУМФ» 2003г. 320 с.
12. М.В.Ким Тезис доклада «Применение сверточный нейронных сетей для распознавания изображений» в сборнике Республиканской научно-технической конференции «Проблемы информационных и телекоммуникационных технологий», проходившей 12-13 марта 2015г. в Ташкенте, Часть 3, с.416-419
13. Селмон Д. «Сжатие данных и изображения и звука» Издательство: Техносфера 2004 г. 368с
14. Хрящёв В. В, Соколенко Е. А., Приоров А. Л. «Сравнение эффективности реализации алгоритмов обучения нейросети в задаче восстановления изображений» Труды II научной конференции «Проектирование инженерных и научных приложений в среде MATLAB» Секция 4. Нейросетевые технологи, стр.1309
15. Иванов М. А. «Применение вейвлет-преобразований в кодировании изображений» стр.169-170
16. Пузий А.Н., Гаврилов И.А. «Межкадровое сжатие видеоданных ТВ изображений на основе оценки и компенсации движения видеообъектов» // «O’zbekiston Respublikasi Qurolli Kuchlari Akademiyasi xabarlari», № 2 (15), часть 1, за ноябрь 2014 г. с.47-54.
17. Электронный учебник по статистике StatSoft «Нейронные сети» http://www.statsoft.ru/home/textbook/modules/stneunet.html

18. П.Г. Круг Учебное пособие по курсу «Микропроцессоры» «Нейронные сети и нейрокопьютеры» Издательство МЭИ,2002г. 177с.
19. Искусственный нейрон https://ru.wikipedia.org/wiki
20. Васенков Данила Валентинович «Методы обучения искусственных нейронных сетей», стр 25
21. Ким М.В. Тезис доклада «Применение нейросетевых технологий для обработки изображений» в сборнике Республиканской научно-технической конференции, проходившей 15-16 марта 2012г. в Ташкенте, Том 3, с.48-50
22. Сергей А.Терехов «Лекции по теории и приложениям искусственных нейронных сетей» Лабораторія Искусственных Нейронных Сетей НТО-2, ВНИИТФ, Снежинск http://alife.narod.ru/lectures/neural/Neu_ch08.htm
23. Пятибратская А.Н «Сеть Хопфилда» http://i-intellect.ru/articles-of-neural-networks/hopfilds-network.html#3
24. Лекция 6 http://apsheronsk.bozo.ru/Neural/Lec6.htm
25. Ф. Уоссермен «Нейрокомпьютерная техника: Теория и практика», Издательство: «Мир»,1992г.,184с.
26. Юрий Борисов, Виталий Кашкаров, Сергей Сорокин «Нейросетевые методы обработки информации и средства их программно-аппаратной поддержки» http://iit.ntu-kpi.kiev.ua/Neuro/LIBRARY/OSP/38.htm#part_6
27. Ким М.В. Тезис доклада «Анализ методов нейросетевого сжатия видеоданных» в сборнике Республиканской научно-технической конференции, проходившей 14 марта 2013г. в Ташкенте, Том 4, с.194-195
28. Изотопов П.Ю., Суханов С.В., Головашкин Д.Л. «Технология реализации нейросетевого алгоритма в среде CUDA на примере распознавания рукописных цифр», 2010год.
29. Википедиа «Сверточная нейронная сеть» https://ru.wikipedia.org/wiki/Свёрточная_нейронная_сеть

30. Rumelhart, David E.; Hinton, G.E.; Williams, R.J. Learning Internal Representations by Error Propagation In Parallel Distributed Processing, Cambridge: M.I.T. Press, v. 1, p. 318-362 (1986).
31. Довженко А.Ю. « Параллельная нейронная сеть с удаленным доступом на базе распределенного кластера ЭВМ»
32. Fatica, M. CUDA for High Performance Computing – Materials of HPC-NA Workshop 3, January 2009.
33. *NeuroPro* нейронные сети, методы обработки и анализа данных http://www.neuropro.ru/neu1.shtml
34. Нейрокомпьютерные технологии http://www.life-prog.ru/1_32549_neyrokompyuternie-tehnologii.html
35. Нейрокомпьютеры http://dfe.petrsu.ru/koi/posob/optproc/neucom.html
36. Нейросети и нейрокомпьютеры http://de.ifmo.ru/bk_netra/page.php?tutindex=25&index=44&layer=1
37. Ким М.В. «Аппаратная реализация нейронных сетей, применяемых для обработки изображений» в сборнике докладов Республиканской научно-технической конференции молодых ученых, исследователей, магистрантов и студентов «Информационные технологии и проблемы телекоммуникаций», проходившей 14 марта 2013 года в г. Ташкенте, часть 4, с. 194-195
38. ИНТУИТ Национальный открытый университет «Лекция 15. Новые технологии проектирования и анализа систем» http://www.intuit.ru/studies/courses/83/83/lecture/20496?page=7
39. Глава 5. Программное обеспечение http://ole-u.narod.ru/Razdel5.html
40. А.Д. Варламов «Восстановление цвета полутоновых изображений нейронной сетью» Муромский институт (филиал) ГОУ ВПО «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых», г. Муром, УДК 004.932.4

41. Сигнейро - технологии обработки и анализа изображений нейронными сетями http://www.xn----7sbabbbc7aihkfne7cddf3ak8a.xn--p1ai/item3_1.php
42. Транспьютер https://ru.wikipedia.org/wiki/Транспьютер
43. А.А. Талалаев, И.П. Тищенко, В.П. Фраленко, В.М. Хачумов статья в журанле: Исскусственный интеллект и принятие решений «Анализ эффективности применения искусственных нейронных сетей для решения задач распознавания, сжатия и прогнозирования», 2008 год
44. Рециркуляционные нейронные сети http://habrahabr.ru/post/130581/