

# СЖАТИЕ ИЗОБРАЖЕНИЙ НА ОСНОВЕ САМООРГАНИЗУЮЩИХСЯ КАРТ КОХОНЕНА

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

Романюк Н.К.

Марченко В.В. – ассистент кафедры ЭВМ

Постоянное увеличение объемов перерабатываемой информации, в частности графической, требует необходимость её компактного(сжатого) представления. В то же время характерная для мультимедийной информации операция дискретизации изображения и звука позволяет ставить и решать задачу эффективного сжатия информации с потерями. Во многих приложениях, например, при представлении неподвижных изображений, такое сжатие более чем оправданно, ведь наиболее популярный в настоящее время стандарт JPEG зачастую не удовлетворяет возрастающим требованиям функциональности.

В этих условиях перспективным представляется развитие подхода, в основе которого лежит использование искусственных нейронных сетей (ИНС). Возможность применения сети Кохонена для сжатия изображений основывается на классическом векторном квантовании (vector quantization) — способе кластеризации, при котором пространство входов делится на ряд областей, для каждой из которых определяется вектор восстановления.

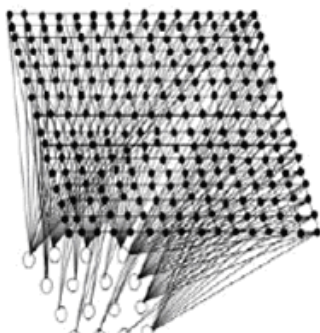


Рис. 1 – Самоорганизующаяся карта признаков

Архитектура самоорганизующейся карты признаков (рис. 1) для векторного квантования определяется размером словаря кодовых векторов. Каждый кодовый вектор — это матрица весов в соревновательном слое. В реализации алгоритма 256 кодовых векторов представляют изображение как выход нейронной сети.

Блок, содержащий 16 пикселей, подается на слой Кохонена, состоящий из 256 нейронов, размещенных в двумерном массиве  $16 \times 16$ . Весовые коэффициенты, связывающие  $j$ -ый нейрон слоя Кохонена и входы сети  $x = \{x_0, x_1, \dots, x_{n-1}\}$ , представлены матрицей  $[\omega_{ij}]$ ,  $j = 0, \dots, 255$ ,  $i = 0, \dots, 15$ .

После того, как словарь создан (нейронная сеть обучена), он может использоваться для сжатия и восстановления изображения по описанной ниже схеме (рис. 2):

- изображение разбиение на блоки  $4 \times 4$ , которые подаются в случайном порядке на вход сети;
- выбирается нейрон, евклидово расстояние от которого до поданного блока  $c = \|\vec{x}^{(i)} - \vec{w}^{(j)}\|^2$  минимально;

• нейроны, чьи веса не победили хотя бы у одного блока, исключаются из словаря;

• множество индексов соответствия нейронов-победителей блокам сжатого изображения сохраняется и кодируется алгоритмом Хаффмана.

При программной реализации алгоритма на языке C# с использованием framework'a AForge.NET для оценки качества сжатия использовались показатели PSNR и MSE. Предложенный алгоритм показал схожие

результаты для различных тестовых образцов. Для иллюстрации полученных результатов используется стандартное тестовое изображение «Lena».

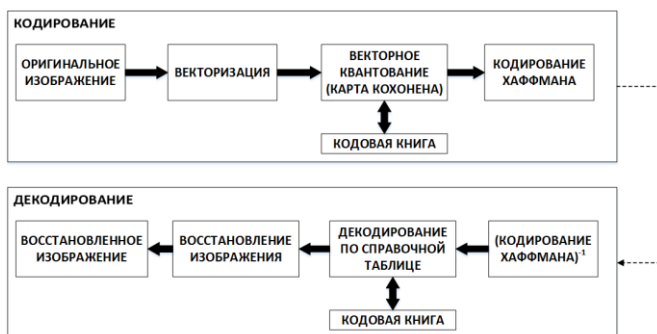


Рис. 2 – Схема кодирования-декодирования изображения

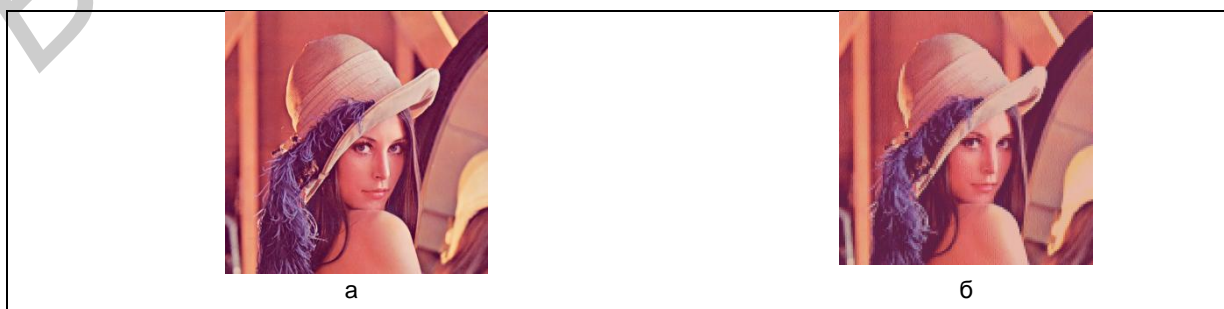


Рис. 3. Изображение: оригинальное (а), сжатое (б)

Таким образом был разработан и проанализирован алгоритм сжатия графики на основе искусственных нейронных сетей. Для создания соответствия между входным и выходным пространствами была применена Самоорганизующаяся карта Кохонена. Результаты моделирования данного алгоритма свидетельствуют о возможности практического применения данного вида ИНС для сжатия изображений.

Список использованных источников:

1. Бодянский, Е.В. Искусственные нейронные сети: архитектуры, обучение, применения. / Е.В. Бодянский, О.Г. Руденко – Харьков 2004. – 369 с.
2. Amerijckx, Christophe. Image Compression by Self-Organized Kohonen Map / С. Amerijckx, M. Verleysen, P. Thissen, J-D. Legat // IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 9, NO. 3 – MAY 1998 – P. 503-507.

## МЕТОДЫ ФОРМИРОВАНИЯ ГИБКОГО ПЛАНА РАБОТ ПО КОНТРОЛЮ КАЧЕСТВА РАЗРАБОТКИ ПС

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

*Новицкий А. А.*

*Насуро Е. В. – к.т.н. доцент*

Повсеместное внедрение программных продуктов и рост сложности используемых технологий и ПС становится причиной роста цены ошибки в продукте, поэтому не теряет актуальности вопрос дальнейшей оптимизации процессов контроля качества (КК) разработки программных средств (ПС). Так как работы по контролю качества разработки востребованы с определения требований к продукту и до самого выпуска, выбор оптимальных для проекта работ по КК отвечает целям повышения надежности итогового продукта, уменьшения издержек, повышения рентабельности процесса разработки, и является актуальной темой для настоящей работы.

Время и бюджет, требуемые на планирование работ и непосредственно сами работы по КК на проекте тем больше, чем сложнее структура и функционал и выше требования к надежности разрабатываемого продукта. Необходимо учитывать и работы по проверке стрессоустойчивости системы, требующие специфической квалификации.

Учитывая многообразие классификаций видов тестирования, в данной работе рассматривается разделение по необходимости выполнения программного кода. Это позволяет четко выделить как минимум одну важную группу работ. По данному критерию методы тестирования делятся на статические, не требующие запуска программного кода, и динамические, непосредственно взаимодействующие с функционирующим приложением или его моделью.

Статические виды тестирования включают работы по верификации требований и анализ программного и скомпилированного кода, вследствие чего могут обеспечивать наиболее раннее выявление различных проблем, потому обязательны к включению в план. [1] Динамические методы тестирования в основном обеспечивают обнаружение дефектов функциональности в ходе проверки тестовых сценариев, или же отвечают задачам проверки устойчивости системы к превышению пределов нормального функционирования, закрытости от влияния извне. Подходы к динамическому тестированию наиболее разнообразны, что создает дополнительные трудности в выборе работ для полного неизбыточного покрытия потребностей разрабатываемого ПС. При этом, различные подходы по различным методам могут быть задействованы на разных этапах жизненного цикла ПС – прямое влияние на это оказывает наращивание функциональности, требующее как полноценной проверки новых фрагментов, так и контроля уже проверенных на предмет регрессии.

Разница в механизме доступа обуславливает специфику проведения тестирования, предпочтительного для каждого из методов. Так как условно-автоматизированное тестирование зачастую сокращает лишь механическую работу тестировщика и может не требовать глубокой экспертизы в сфере автоматизации процесса тестирования, в контексте исследуемой темы допустимо включить данный подход в категорию ручного тестирования. Таким образом, рассматривать следует критерии эффективности ручного и автоматизированного тестирования.

Независимо от того, будет ли реализовано динамическое тестирование посредством ручного или автоматизированного подхода, общим вариантом критерия оценки покрытия является покрытие, основанное на спецификации или требованиях. Главное требование состоит в покрытии некоторого утвержденного минимума – набора требований.

При том, что механизм планирования и оценки покрытия, разрабатываемого ПС тестированием может быть выбран в соответствии с потребностями системы и имеющейся документации, выбор применяемых методов и техник строго не регламентируется. План, методы и техники могут определяться командой на этапе планирования исходя из субъективных мнений, предпочтений, доступных ресурсов. [3] Таким образом, задача формирования полного неизбыточного плана работ по КК разработки, пригодного для внедрения в проекты,