

Юденков В. С., доцент (БГТУ);  
Винобер А. Ю., магистрант (БГТУ)

## АЛГОРИТМ АДАПТИВНОГО РАСТРИРОВАНИЯ ЦИФРОВЫХ ИЗОБРАЖЕНИЙ НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ

Целью данной работы является создание адаптивного растрового процессора, который позволит устранить недостатки существующих растровых процессоров и улучшить качество печати.

The aim of my research is to create the adaptive raster image processor, which can correct the drawbacks of existing RIPs and to improve the quality of printing.

**Введение.** В памяти компьютера изображения хранятся в цифровом виде и состоят из пикселей. Процесс преобразования изображения в последовательность точек голубого, пурпурного, желтого и черного цветов (СМΥК) носит название растривания.

В профессиональной печати используются специализированные RIP. Существует большое количество способов печати. Один из них — прямая цифровая печать, которая является очень популярным современным методом печати в полиграфии. Установленный на цифровой печатной машине RIP может иметь недостатки. И полностью заменить его практически невозможно. Тогда есть выход — использование другого растрового процессора.

Целью исследования является создание программного адаптивного растрового процессора, который будет осуществлять растривание с использованием различных технологий печати в зависимости от того, какой оригинал необходимо напечатать. Это позволит улучшить качество печати и снизить время растривания.

Как показывает практика, применение определенных способов растривания для конкретного оригинала приводит к улучшению качества печати. Так, наилучшие результаты для сюжетов «кожи» и «светов» получаются с иррациональным растриванием, а для теней — со стохастическим растриванием. Круглые точки часто используются для печати фотоснимков продукции, эллиптические — для сюжетов с людьми, а квадратные — для тем, которые требуют четкого рисунка. Круглые или эллиптические точки обычно лучше всего подходят для черно-белой печати; эллиптические — для цветной печати. Значит, определив тип оригинала, можно применить нужный алгоритм растривания либо выбрать форму точки и получить на выходе печатный оттиск требуемого качества.

**Методы сравнения изображений.** Растровый процессор меняет способ растривания в зависимости от того, какой оригинал [1]. Тип оригинала определяется следующим образом: существует база данных усредненных изображений, если сравнить исходное изображение и существующее в базе, то можно на основе определенных алгоритмов сделать вывод об их схо-

жести или расхождении. К таким методам сравнения относятся: метод вычитания, метод на основе критерия PSNR, гистограммный метод, нейросетевой метод.

**Метод вычитания.** Наиболее простой алгоритм сравнения двух изображений — это обычное вычитание, взятое по модулю, оно осуществляется функцией Matlab *imabsdiff* [2], которая имеет входными параметрами два изображения.

При использовании оператора сравнения возвращается матрица, состоящая из нулей и единиц: нули на месте, где не выполняется условие — false, и единица на месте, где выполняется — true. Число 75 выбрано опытным путем при построении гистограммы изображения, т. к. при таком значении цвета довольно близки к черному. При достаточно большом результате можно сказать, что изображения совпадают.

**Метод на основе критерия PSNR.** Следующим методом является сравнение двух матриц изображений при помощи критерия PSNR (peak signal-to-noise ratio) [2]. Этот критерий определяется по формуле

$$psnr = 20 \cdot \lg\left(\frac{N_{\max}}{\sqrt{\text{Mean}(\text{Mean}((A - B)^2))}}\right). \quad (1)$$

Пример программы в Matlab для использования критерия PSNR:

```
if A == B
error('Images are identical: PSNR has infinite value')
end
max2_A = max(max(A));
max2_B = max(max(B));
min2_A = min(min(A));
min2_B = min(min(B));
if max2_A > 255 || max2_B > 255 || min2_A < 0 || min2_B < 0
error('input matrices must have values in the interval [0,255]')
end
error_diff = A - B;
dbels=20*log10(255/(sqrt(mean(mean(error_diff.^2)))));
```

Оригинал сравнивается с базой усредненных изображений для каждого сюжета. Далее создается небольшая программа *wii* (what is it). Она выделяет наименьшее значение критерия PSNR, и на

основе этого делается вывод о принадлежности оригинала к определенному типу изображения.

**Гистограммный метод.** Для построения гистограммы необходимо: загрузить изображение в Matlab и выполнить команду plot (вектор-строка) [2]. Для получения вектор-строки из матрицы изображения следует выполнить команду sum в зависимости от того, каким образом нужно проанализировать изображение: по вертикали или по горизонтали.

Для анализа построенных гистограмм можно воспользоваться следующим способом: подсчитать значение математических величин, которые характеризуют случайные величины, такие как математическое ожидание, дисперсия, среднеквадратичное отклонение.

Математическое ожидание определяется по формуле

$$mx = \frac{1}{n} \cdot \sum_{k=1}^n x_k, \quad (2)$$

где  $n$  — количество элементов вектор-строки или вектор-столбца;  $x_k$  — значения яркости.

Дисперсию можно найти по формуле

$$dx = \frac{1}{n} \left( \sum_{i=1}^n (x_i - mx)^2 \right), \quad (3)$$

где  $mx$  — математическое ожидание.

Среднеквадратичное отклонение определяется как

$$\sigma = \sqrt{dx}. \quad (4)$$

Используя выражения (2)–(4), получаем значения, указанные в таблице. Первая и вторая строки относятся к классу изображения — портрет, остальные — природа. Из сравнения строк таблиц явно видна разница. Так, для портрета значения математического ожидания находятся в пределах  $2\text{--}2,8 \cdot 10^5$ , а для других значительно меньше. Кроме того, дисперсия тоже отличается на порядок.

Из анализа практических испытаний вышеприведенных методов лучшие показатели имеет критерий сравнения PSNR.

**Сегментация изображений.** Сегментация делит изображение на составные части и объекты.

Степень детализации этого деления зависит от решаемой задачи, т. е. сегментацию следует остановить, когда интересующие объекты уже выделены или изолированы. Сегментация изображений является довольно непростой задачей обработки изображений. Точность сегментации во многом определяет успех конечных компьютеризованных процедур анализа изображений. Как правило, алгоритмы сегментации монохромных изображений основаны на одном из двух базовых свойств яркости изображения: разрывности и однородности. В первом случае подход состоит в разбиении изображения на части исходя из резких перепадов значений яркости, которые происходят, например, на границах объектов. Вторая группа методов осуществляет разделение изображений на области, однородные в смысле определенных, заранее заданных критериев.

В данном случае возможно обнаружение резких яркостных перепадов изображения для выделения конкретных частей оригинала и определения, к какому сюжету данная часть относится. Это действие возможно осуществить при помощи функции edge (изображение, 'метод', параметры) [2]. Метод — способ оценки яркостного перепада (детектор краев) изображения. Существует несколько методов [2]. В данной статье в ходе практического исследования выяснилось, что метод Канни наиболее эффективен.

**Нейросетевые методы распознавания изображений [4].** Основные направления применения различных НС для распознавания образов и изображений:

- применение для извлечения ключевых характеристик или признаков заданных образов;
- классификация самих образов или уже извлеченных из них характеристик (в первом случае извлечение ключевых характеристик происходит неявно внутри сети);
- решение оптимизационных задач.

Как правило, под «обычными», или «классическими», нейросетями понимают полносвязные нейронные сети прямого распространения с обратным распространением ошибки. Как следует из названия, в такой сети каждый нейрон связан с каждым, сигнал идет только в направлении от входного слоя к выходному, нет никаких рекурсий.

### Значения математических величин

Математическое ожидание	Дисперсия	Среднеквадратичное отклонение
$mx2 = 2,1234e + 005$	$dx2 = 7,6564e + 011$	$sigma2 = 8,7501e + 005$
$mx3 = 2,6321e + 005$	$dx3 = 1,3452e + 012$	$sigma3 = 1,1598e + 006$
$mx4 = 1,1347e + 005$	$dx4 = 2,3327e + 013$	$sigma4 = 4,8298e + 006$
$mx5 = 1,2087e + 005$	$dx5 = 2,0555e + 013$	$sigma5 = 4,5338e + 006$
$mx6 = 1,2636e + 005$	$dx6 = 1,8613e + 013$	$sigma6 = 4,3143e + 006$
$mx7 = 1,1926e + 005$	$dx7 = 2,1146e + 013$	$sigma7 = 4,5985e + 006$
$mx8 = 8,2992e + 004$	$dx8 = 3,6591e + 013$	$sigma8 = 6,0490e + 006$
$mx9 = 9,5393e + 004$	$dx9 = 3,0836e + 013$	$sigma9 = 5,5530e + 006$

Имеется два блока. На первом этапе происходит сравнение оригинала с базой изображений с известным типом. Сама база до конца не создана, есть ряд тестовых изображений. Сравнение производится с помощью нейросети. Моделирование осуществляется на базе пакета Matlab, который имеет встроенные возможности создания нейросетей с использованием библиотечных функций.

Наиболее чувствительным фрагментом изображения к методу растривания является лицо человека. Поэтому на следующем этапе производится поиск фрагментов типа лица, если таковые присутствуют. Далее подсчитывается отношение площади найденного лица к общей площади оригинала. При превышении определенного значения, задаваемого вручную, выбираются способ и параметры растривания.

Достоинства и недостатки. Нейросетевые методы обеспечивают быстрое и надежное распознавание изображений. Однако при применении этих методов к изображениям трехмерных объектов возникают трудности, связанные с пространственными поворотами и изменением условий освещенности.

Данное направление по распознаванию образов и классификации изображений динамично развивается и находит свое применение не только в полиграфии. Выбор метода распознавания зависит от конкретной задачи. Метод вычитания требует наименьших вычислений и, следовательно, минимального отрезка времени на обработку данных, но уровень точности довольно низок и не может быть достаточен для работы адаптивного растрового процессора без участия оператора. Гистограммный метод и метод по критерию PSNR наиболее оптимальны, т. к. дают приличный уровень точности, при этом имеют довольно простой способ реализации и не требуют больших математических вычислений. Нейросетевой метод наиболее сложный и самый точный. Для составного изображения следует применять сегментацию или выделение однородных участков цвета, а затем для отдельных участков использовать методы сравнения.

Блок-схема работы программы по распознаванию типа изображения представлена на рис. 1.

**Разработка алгоритма адаптивного растрового процессора.** Для моделирования растрового процессора использовался пакет MATLAB версии 6.5. Эта программа позволяет эффективно решать задачи, возникающие в области обработки цифровых изображений.

Редактор Matlab — это одновременно и специализированный текстовый редактор для создания М-файлов. Они являются скриптами, т. е. состоят из последовательностей исполняемых команд Matlab, или функциями, которые допускают использование аргументов и имеют собственные выходные данные.

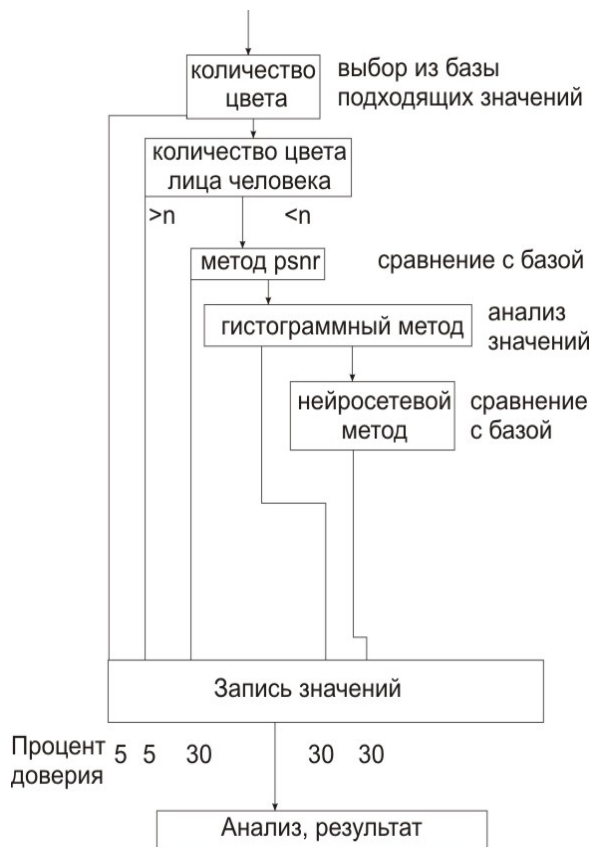


Рис. 1. Схема работы программы по распознаванию типа изображения

В процессе моделирования определяется тип оригинала, а затем для идентифицированного оригинала применяется либо традиционный алгоритм растривания с углами поворота раstra, либо стохастическое растривание. Перед растриванием проводится операция цветоделения [5]. После растривания происходит запись готового файла на жесткий диск.

Подготавливается тестовое изображение. Программа сама изменяет его размер на 1600×1200, поскольку все измерения проводились для изображений именно такого размера. Для изменения размеров используется стандартная функция `imgrescale` (изображение [размер вертикальный, горизонтальный]) [2]. Далее с помощью этой же функции размер входного изображения уменьшается в 4 раза, т. к. обработка большого изображения требует длительного времени и достаточного количества памяти компьютера, при размере 1600×1200 объем памяти должен быть очень большим. После подсчета математических величин можно задавать все необходимые условия, в данном случае задается диапазон значений математического ожидания. Функции `rastr_stoh`, `rastr`, `rastr_15`, `rastr_75` и `rastr_45` — это функции растривания.

```
>> a=imread('metr_test.jpg');
```

```
>> [c,m,y,k]=adopt(a,1).
```

Получается четыре массива — четыре отраstrиванных изображения для каждого цвета.

Алгоритм разработанного растрового процессора представлен на рис. 2.

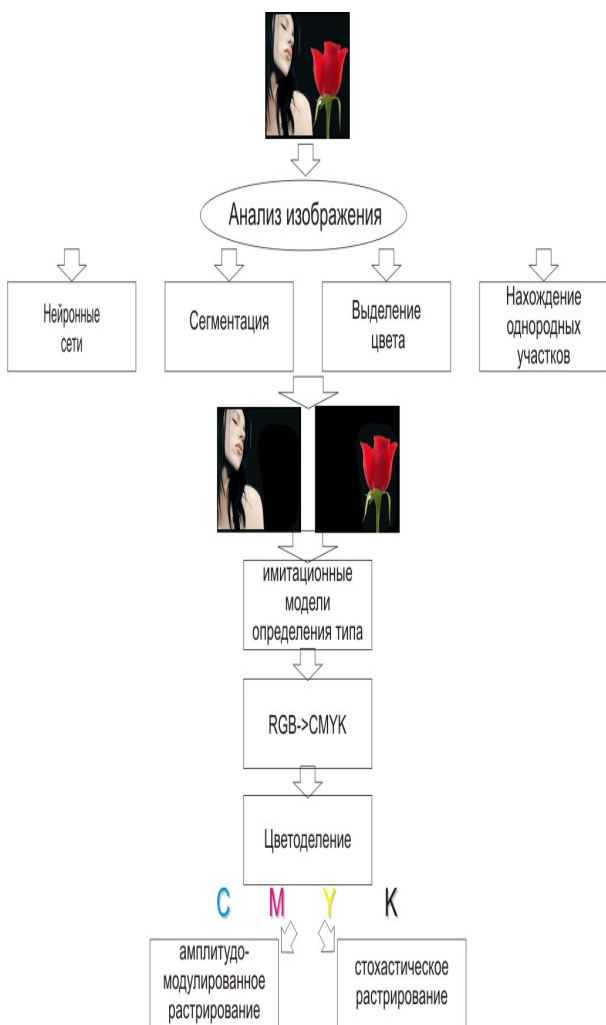


Рис. 2. Алгоритм адаптивного растрового процессора

**Формирование точки.** Рассмотрим использование заполнения по-спирали [5].

На рис. 3 представлена растровая ячейка с 3-процентным заполнением.

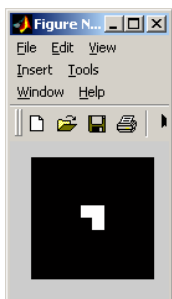


Рис. 3. Ячейка с 3-процентным заполнением

**Моделирование стохастического алгоритма растрового процессора в среде Matlab.** Стохастический алгоритм растрирования получается следующим образом: используется

стандартная функция Matlab под названием `Randperm` [2], которая использует встроенный генератор случайных чисел и формирует массив, состоящий из натуральных чисел от 1 до заданного значения (`max_value`), распределенных случайным образом. Формируется вектор-строка случайных чисел, количество которых равно размеру растровой ячейки в квадрате. Каждое число вектора — координата следующей точки. Таким образом, проводя простое преобразование, можно вычислить номер строки и номер столбца матрицы. Получается набор координат заполнения точками растровой ячейки. В зависимости от процента заполнения зарисовывается определенное количество точек ячейки.

**Проведение экспериментальных исследований.** Местом проведения экспериментальных исследований был выбран Объединенный институт проблем информатики НАН Беларуси [5].

В ОИПИ НАН разработаны фотоплоттеры планшетного (ФП-2) и барабанного (РФП) типов [6]. Они имеют апертуру записывающих пучков  $10 \times 10$  мкм и обеспечивают, при таких же значениях накопленных погрешностей, вывод фотоформ с максимальным разрешением 2540 dpi.

Экспонирование производилось на планшетном фотоплоттере с разрешением 1270 dpi, что соответствует размеру точки 20 мкм. Был подготовлен монохромный оригинал, который отрастрирован адаптивным растровым процессором и растровым процессором Harlequin. Растрированные изображения сохранены в формате *tiff*. После экспонирования получены фотоформы. На их основе в типографии БГТУ изготовлены печатные формы плоской офсетной печати и на печатной машине «Доминант» получены печатные оттиски (рис. 4, 5, 6).

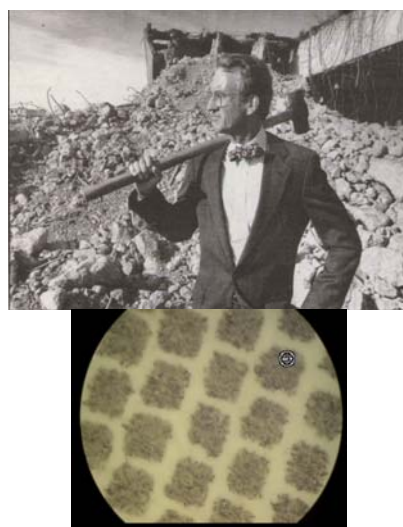


Рис. 4. Оттиск 1, квадратная форма точки

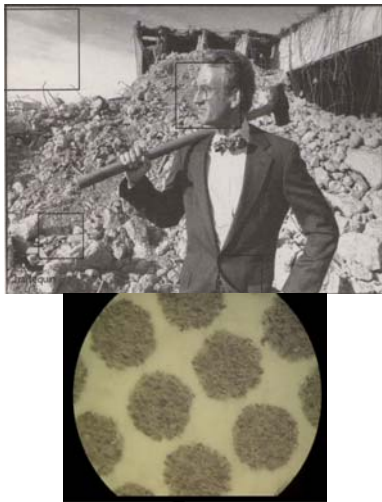


Рис. 5. Оттиск 2, круглая форма точки



Рис. 6. Оттиск 3, стохастическое растривание

Визуальный анализ отпечатков показал, что с использованием стохастического алгоритма растривания получается наилучшее качество при печати мелких деталей и печати лица человека, однако в светах происходит потеря мелких деталей, при этом в области теней потери наиболее незначительны. При использовании стандартного алгоритма растривания не происходит потери деталей в области светов, но при этом качество ниже. Стандартный алгоритм растривания с применением растрового процессора Harlequin RIP-kit [7] показал несколько лучшие результаты по ка-

честву, чем такой же собственный стандартный алгоритм. При этом с использованием круглой точки детали в области теней передаются лучше. Тем не менее качество печати с применением стохастического алгоритма растривания выше.

**Вывод.** Разработаны методы определения типа изображения, основанные на использовании алгоритмов вычитания матриц изображений, гистограммного анализа, вычисления критерия PSRN, построения нейросетей. В результате создан прототип программного адаптивного растрового процессора, который был проверен на практике и позволил улучшить качество печатных оттисков.

### Литература

1. Винобер, А. Ю. Применение компьютера для растривания изобразительной информации / А. Ю. Винобер // 58-я студенческая науч.-техн. конф.: материалы докл.: в 3 ч. — Минск: БГТУ, 2007. — Ч. 3. — С. 242–243.
2. Гонсалес, Р. Цифровая обработка изображений в среде MATLAB / Р. Гонсалес. — М.: Изд-во Техносфера, 2006. — 616 с.
3. Винобер, А. Ю. Повышение качества растривания цифровых изображений при прямой цифровой печати / А. Ю. Винобер // Студенческая наука — будущее государства: в 2 ч. — Пинск: ПолесГУ, 2008. — Ч. 1. — С. 62–63.
4. Брилюк, Д. В. Нейросетевые методы распознавания изображений / Д. В. Брилюк. — 78-я науч.-техн. конф.: материалы докл.: в 4 ч. — Минск: ОИПИ НАН Беларуси, 2001. — Ч. 3. — С. 242–243.
5. Винобер, А. Ю. Адаптивная система управления процессором растривания изобразительной информации для прямой цифровой печати / А. Ю. Винобер // 59-я студенческая науч.-техн. конф.: материалы докл.: в 3 ч. — Минск: БГТУ, 2008. — Ч. 3. — С. 374–377.
6. Ткаченко, В. В. Системы записи полиграфических фотоформ с использованием светодиодных матричных модулей (LED-головок) / В. В. Ткаченко // Труды БГТУ. Сер. IX, Издат. дело и полиграфия. — 2009. — Вып. XVII. — С. 40–44.
7. Винобер, А. Ю. Повышение качества флексографской печати на базе адаптивного растрового процессора / А. Ю. Винобер // 60-я студенческая науч.-техн. конф.: материалы докл.: в 3 ч. — Минск: БГТУ, 2009. — Ч. 2. — С. 132–134.

Поступила 29.06.2010