

Рисунок 1 - Изображение исследуется полностью

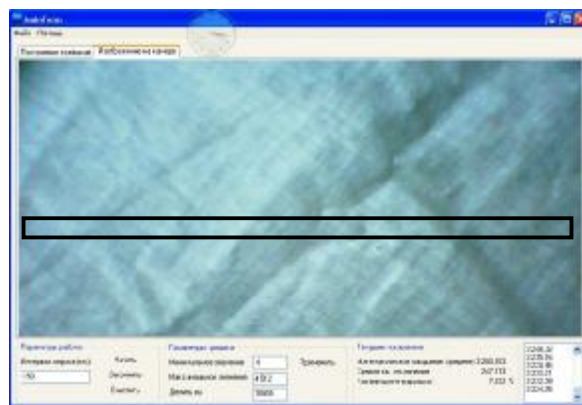


Рисунок 2 - Исследуется сектор изображения

Подобный подход может позволить выявить бракованные участки вдоль полотна. Если у нас имеется более мощный компьютер, то выделенную область мы можем разбить на сектора и анализировать уже полотно и поперёк – по утку. Таким образом можно контролировать качество полотна как на протяжении всей партии, так и отслеживать брак в конкретном месте по ширине полотна.

УДК 677.017.35

ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ АЛГОРИТМОВ ВЫДЕЛЕНИЯ РАПОРТОВ ИЗ ТОЧЕЧНЫХ ИЗОБРАЖЕНИЙ КЛАССА 1 МЕТОДОМ СТАТИСТИЧЕСКОГО МОДЕЛИРОВАНИЯ

Г.И. Борзунов

ГОУ ВПО «Московский государственный текстильный университет
имени А.Н. Косыгина», г. Москва, Российская Федерация

Изображения класса 1 характеризуются точным повторением раппорта без каких-либо изменений кодов отдельных точек. Обычно такие изображения получаются в результате повторения некоторого раппорта в среде графического редактора и распространяются на различных носителях информации или с помощью сетевых технологий. В предыдущих публикациях были предложены алгоритмы выделения из изображений класса 1 раппортов: медленный простейший алгоритм Rpt1 /1/ и быстрый алгоритм Rpt3 /2/. В этих же работах были даны оценки алгоритмов Rpt1б Rpt3 в лучшем и худшем случаях. В данном докладе приводятся результаты вычислительного эксперимента по определению временной сложности алгоритмов Rpt1, Rpt3, в ситуации, которую в теории анализа алгоритмов принято называть средним случаем. При проведении эксперимента использовались реализации указанных алгоритмов на языке программирования Visual C#. При программировании указанных алгоритмов для работы с графическими изображениями использовались стандартные функции из библиотеки *System.Drawing*, а именно, *Bitmap.GetPixel* и *Bitmap.SetPixel*. В начале работы изображение переносилось в массив $A[m][n]$ с многократным использованием функции *Bitmap.GetPixel*. Для отображения полученного раппорта использовалась функция *Bitmap.SetPixel*.

В данной работе используются следующие обозначения: $A[K][K]$ – массив, в котором размещается точечное ахроматическое изображение, т.е. элементы A принимают значения в

пределах от 0 до 255; N – количество раппортов, которые укладываются в исходном изображении;

Вычислительный эксперимент проводился с помощью компьютера, имеющего следующие системные параметры: процессор - Pentium(R)4 с тактовой частотой 2,00 ГГц (CPU 2,00 GHz), объём оперативной памяти (ОЗУ) - 512 МБ, операционная система - MS XP. При проведении вычислительного эксперимента использовались ахроматические точечные изображения, генерация которых выполнялась следующим образом. Осуществлялся ввод размерности точечного изображения K и количества раппортов в этом изображении N . Распределялась память для массива $A[K][K]$, предназначенного для размещения и хранения кодов точек изображения. После этого левая верхняя область этого массива $A[K/N][K/N]$, размер которой совпадал с раппортом генерируемого изображения, заполнялась случайными числами из диапазона от 0 до 255, определяющими интенсивность серого цвета. Далее копиями этой области заполнялся весь массив $A[K][K]$. На этом заканчивалось построение изображения класса 1.

Для сравнительного анализа экспериментальной временной сложности алгоритмов Rpt1, Rpt3 в среднем случае были выполнены соответствующие измерения числа сравнений и времени работы процессора при выделении раппортов. Выделение раппортов в изображениях первого класса, размерность которых менялась от 10×10 вплоть до 1200×1200 , выполнялось достаточно быстро (менее чем за 1 сек). Например, для выделения раппортов размерностью 12×12 ($N=100$) в изображениях, размерность которых была равна 1200×1200 , при параметре $Noise=0,02$ в среднем требовалось: 12384930 сравнений (0,9064063 сек) при использовании алгоритма Rpt1 и 11545072 сравнений (0,8767187 сек) при использовании алгоритма Rpt3. Несмотря на весьма малые временные затраты на решение рассматриваемой задачи, следует отметить необходимость рационального выбора между алгоритмом Rpt1 и алгоритмом Rpt3, когда выделение раппорта является функцией внешнего алгоритма обработки изображений, вложенной во внутренний многократно повторяющийся цикл (например, при повторении цикла 10^6 раз). В этом случае если разница во времени однократного выделения раппорта составляет 0,03 секунды, то при многократном решении этой задачи суммарная разница временных затрат будет равна $0,03 \times 10^6 = 3 \times 10^4 \text{ с} = 8,333333 \text{ час}$. В таблице 1 приводятся результаты исследования зависимости числа сравнений (временной сложности) при выделении раппортов из изображений класса 1 от возрастающих значений размерности точечного изображения (K) при количестве раппортов в изображении равном 9 ($N=9$). При сравнении данных, приведённых в таблице 1 видно, что при возрастании размерности точечного изображения (K) от 50 до 500 превышение числа сравнений при использовании алгоритма Rpt1 над числом сравнений при применении алгоритма Rpt3 возрастает от 16,9% до 65,1%.

Таблица 1

К	Rpt1	Rpt3
	Число сравнений	Число сравнений
50	12976	11097
100	103566	74398
150	353289	237360
200	823261	534808
250	1626660	1032545
300	2837366	1773100
350	4469778	2763357
400	6715848	4116176
450	9619907	5856389
500	13122519	7946662

Однако, при малых размерностях раппорта и сравнительно небольших размеров изображения соотношение может оказаться обратным. Так при $N=225$ и $K=500$ (размерность раппорта 2×2) при использовании алгоритма Rpt1 для выделения раппорта потребовалось 606190 сравнений, а при использовании алгоритма Rpt3 раппорт был выделен после выполнения 829026 сравнений. Но и в этом случае при дальнейшем увеличении размеров изображения K временная сложность Rpt3 оказывается меньше временной сложности Rpt1. В качестве экспериментального подтверждения этого положения в таблице 2 приводятся значения параметра K , начиная с которых алгоритм Rpt3 становится и остаётся быстрее алгоритма Rpt1 при выделении раппортов размерностью 4×4 , 6×6 (в таблице 2 параметр R).

Таблица 2

K		20	280	800
N		5	60	100
R		4	4	8
T(K)	Rpt1	1758	382320	7320943
	Rpt3	1743	378888	7006807

Графики, представленные на рисунке 1 также подтверждают, что в области $K > 80$ при увеличении K временная сложность алгоритма Rpt3 становится меньше временной сложности алгоритма Rpt1, и разница между ними непрерывно увеличивается. В этом эксперименте, результаты которого представлены в виде этих графиков, вычисления повторялись 100 раз в каждой точке при $N=10$. Таким образом, данные вычислительного эксперимента подтверждают теоретические оценки временной сложности алгоритмов Rpt1, Rpt3 /1,2/: алгоритм Rpt3 выделяет раппорты изображений класса 1 быстрее по сравнению с алгоритмом Rpt1, и это преимущество растёт с увеличением размерности анализируемого изображения.

Зависимость числа сравнений T(K) от размеров изображений K

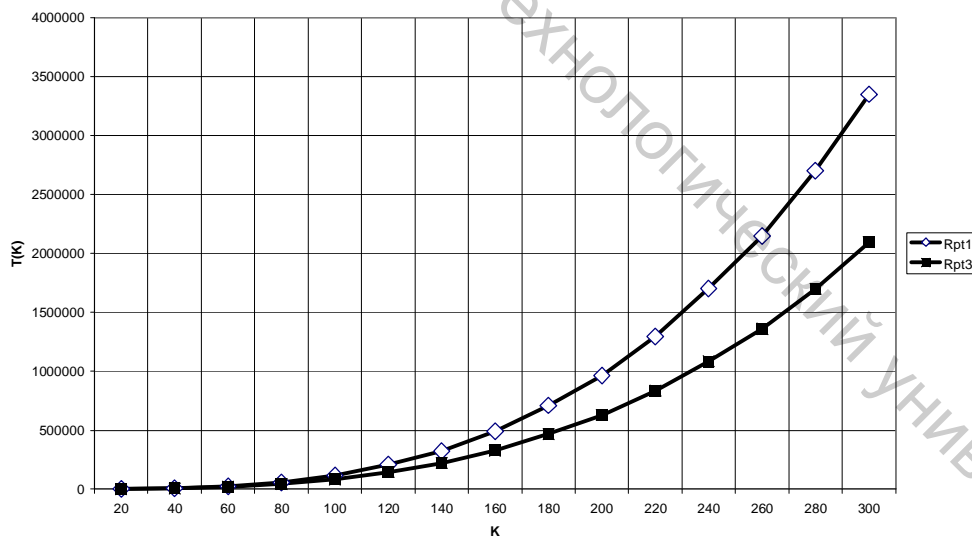


Рисунок 1

Список использованных источников

1. Borzunov G. I. Study of algorithms for recognition of pixel image rappers// Proceedings of higher education institutions. Textile industry technology. – 2009, # 3С, – с. 101 .. 103.
2. Борзунов Г.И., Моисеев К. А. Быстрое выделение раппорта точечного рисунка без учёта возможных помех. Современные технологии и оборудование текстильной промышленности (ТЕКСТИЛЬ 2008), 11 – 12 ноября, 2008. Г. Москва, ГОУВПО «МГТУ им. А. Н. Косыгина».