

## СРАВНЕНИЕ РАСТРОВОЙ ОБРАБОТКИ ИЗОБРАЖЕНИЯ И ПРЕДСТАВЛЕНИЯ ПРОСТРАНСТВЕННЫХ ОБЪЕКТОВ ОТРЕЗКАМИ КРИВЫХ

**Аннотация:** В статье рассматривается представление изображений в виде растровой развертки и в виде пространственных объектов отрезками кривых, заполняющих растровое пространство.

**Ключевые слова:** растровое изображение, объекты, отрезки, кривые, развертка раstra.

**Annotation:** The article considers the representation of images in the form of raster scanning and in the form of spatial objects by segments of curves filling the raster space.

**Key words:** raster image, objects, segments, curves, raster scanning.

Построчной или прогрессивной растровой разверткой называется развертка раstra в непрерывной последовательности строк: 1-я, 2-я, 3-я и так далее (рисунок 1).

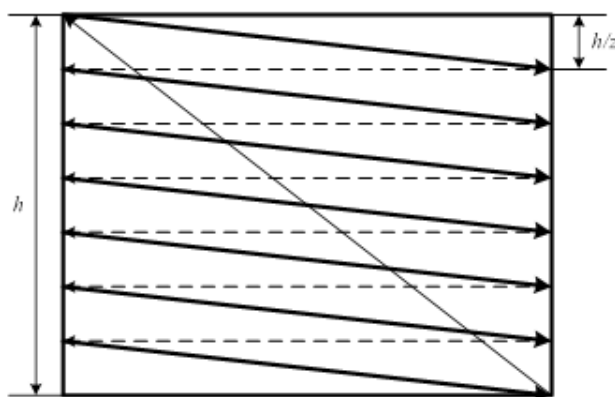


Рисунок 1. Построчная растровая развертка

Развертывающий элемент движется по горизонтали с постоянной скоростью, прочерчивая строку растра, и одновременно смещается по вертикали, так что к исходу строки он сместится вниз относительно ее начала на  $h/z$ , где  $h$  – высота растра,  $z$  – количество строк в растре, то есть на ширину одной строки. Быстро возвращаясь к началу строки, развертывающий элемент займет положение, соответствующее началу второй строки и так далее [1-4].

При построчной растровой развертке изображения данные в пространственную базу данных заносятся следующим образом Rstr (num\_x, num\_y, num\_l), где num\_x – координата «x» начала отрезка, num\_y – координата «y» начала отрезка, num\_l – длина отрезка в количестве ячеек растрового пространства. Пример занесения полученной информации в базу данных PostgreSQL представлен на рисунке 2.

```

try {
    for(int i=1;i<=num.j++){
        ArrayList<Line> tempLine = lineList.get(i-1);
        int le = tempLine.size();
        for(int j=0;j<le;j++){
            int tx = tempLine.get(j).getX();
            int ty = tempLine.get(j).getY();
            int tl = tempLine.get(j).getL();
            String sx = Integer.toBinaryString(tx);
            String sy = Integer.toBinaryString(ty);
            String sl = Integer.toBinaryString(tl);
            Statement statement = null;
            statement = connection.createStatement();
            String insertTableSQL = "INSERT INTO public.\"Method1\"(num_x, num_y,
num_l)VALUES ('"+sx+"','"+sy+"','"+sl+"');";
            statement.executeUpdate(insertTableSQL);
            statement.close();
            //s = s+Integer.toBinaryString(tx)+" "+Integer.toBinaryString(ty)+"
"+Integer.toBinaryString(tl)+" "+n"; }
            connection.close();
        } catch (SQLException ex) {
            Logger.getLogger(Main.class.getName()).log(Level.SEVERE, null, ex);
        }
        System.out.println("DONE");
    }
}

```

## Рисунок 2. Занесение полученной информации в базу данных PostgreSQL

С целью уменьшения затрат памяти используем новый метод представления данных для сохранения в пространственную базу данных [5-10]. использовать форматы: Rstr (Object, Code). Элемент Code содержит в себе информацию об адресе начала отрезка и о количестве ячеек в этом отрезке.

Элемент Code формируется по следующим правилам [5].

— Если Code указывает на отрезок из одной ячейки, то знаковый бит равен нулю, а остальные биты в Code совпадают с адресом начала отрезка кривой, заполняющей пространство.

— Если Code указывает на отрезок из двух и более ячеек, то знаковый бит равен единице, а остальные биты в Code первоначально совпадают с адресом начала отрезка кривой, заполняющей пространство, но при дальнейшем выполнении процедуры формирования Code могут изменить нулевое значение на единичное. Изменение может коснуться только бит составляющих последовательность подряд идущих нулей, включающую младший бит в адресе начала отрезка кривой, заполняющей пространство.

— Адрес начала отрезка кривой четный.

— Количество ячеек в отрезке кривой – четное, но может принимать только те значения степени двойки, которым кратен адрес начала отрезка.

Если это количество равно максимальному значению, то младший бит в Code остается равным нулю. В противном случае он устанавливается в единицу и дополнительно в единицу устанавливается бит в позиции кода, после которой остается количество нулей равное степени двойки, определяющей количество ячеек в отрезке.

Отсюда следует, что при представлении любого отрезка новым методом, мы получаем выигрыш в количестве используемого объема памяти. Объем используемой памяти в новом методе в два раза меньше, чем в методе растровой развертки.

#### **Использованные источники:**

1. Атакищев О.И. Отображение графической и атрибутивной информации фрагментов изображения, представленных линейными квадродеревьями, на основе операция реляционной алгебры [Текст] / О.И. Атакищев, А.В. Белов, В.Г. Белов / Научоемкие технологии. 2012. Т. 13. № 9. С. 34-37.

2. Белов А.В. Представление квадродеревьев бинарными деревьями [Текст] / А.В. Белов, Т.М. Белова // Известия Юго-Западного государственного университета. Серия: Управление, вычислительная техника, информатика. Медицинское приборостроение. 2013. № 1. С. 12-15.

3. Белов А.В. Способы хранения растровых данных на основе квадродеревьев в системах поддержки принятия решений [Текст] / А.В. Белов, Т.М. Белова // Известия Юго-Западного государственного университета. 2012. № 4-2 (43). С. 84-87.

4. Белова Т.М. Структура программы для представления алгоритмов управления процессом тестирования с помощью структуры данных [Текст]/ Т.М. Белова, В.Г. Белов, К.А. Жерденко // Информационные системы и технологии: материалы докладов II Международной научно-практической заочной конференции «ИСТ -2016». – Курск, ЗАО «Университетская книга», 2016. – С. 52 -54.

5. Белов В.Г. Представление пространственных объектов отрезками кривых, заполняющих растровое пространство [Текст] / В.Г. Белов, Т.М. Белова // Интеллектуальные информационные системы: тенденции, проблемы, перспективы. Материалы докладов IV международной заочной научно-практической конференции «ИИС-2016» (20 января 2017 г.). – Курск: ЗАО «Университетская книга», 2017. – С. 8 -11.

6. Белова Т.М. Представление параллельных и асинхронных алгоритмов в виде структур данных [Текст] / Т.М. Белова, Е.С. Кофанова, А.С. Тулупцева // Интеллектуальные информационные системы: тенденции, проблемы, перспективы. Материалы докладов IV международной заочной научно-практической конференции «ИИС-2016» (20 января 2017 г.). – Курск: ЗАО «Университетская книга», 2017. – С. 11 -12.

7. Белов В.Г. Способ кодирования для растровой формы представления пространственных объектов [Текст] / В.Г. Белов, Т.М. Белова // Оптико-электронные приборы и устройства в системах распознавания образов,

обработки изображений и символьной информации. Распознавание 2017. Сборник материалов XIII Международной научно-технической конференции (16 – 19 мая 2017 г.). – Курск: ЗАО «Университетская книга», 2017. – С. 63-64.

8. Атакищев, О.И. Трехуровневая объектно-ориентированная модель организации параллельных асинхронных вычислительных процессов в ГИС [Текст] / О.И. Атакищев, Т.М. Белова, М.В. Белов // Известия Курск. гос. техн. ун-та. - 2004. - №2(13). - С. 67-72.

9. Белов В.Г. Определение пересечения пространственных объектов, представленных в растровой форме, с помощью модифицированных В PLUS деревьев [Текст] / В.Г. Белов, Т.М. Белова // Информационные системы и технологии. Сборник материалов III Международной научно-технической конференции. – Курск: ЗАО «Университетская книга», 2017. – С. 56 - 58.

10. Белов В.Г. Определение пересечения пространственных объектов, представленных в растровой форме, с помощью операции естественного соединения [Текст] / В.Г. Белов, Т.М. Белова // Инфотелекоммуникации и космические технологии: состояние, проблемы и пути решения: сборник научных статей по материалам I Всероссийской науч.-практ. конф.: в 2 ч. – Ч. 1 / редкол.: В. Г. Андронов (отв. ред.) [и др.]; Юго-Зап. гос. ун-т. – Курск, 2017. – С. 333 – 335.