

4. Демичев Д.А. Виды рекламы в Интернете. Экономические науки 2019. С. 1-6.

5. Курбанов М.М. Роль и эффективность рекламы. Экономика и социум №6 ч.2. 2021. С. 794-800.

УДК 004.925

Н.И. Уласевич, Н.А. Жилияк

Белорусский государственный технологический университет
Минск, Беларусь

ПРЕОБРАЗОВАНИЕ РАСТРОВЫХ ИЗОБРАЖЕНИЙ В SVG НА ОСНОВЕ ЭЛЕМЕНТА ЭЛЛИПС

***Аннотация.** В статье рассматриваются возможности преобразования изображений в SVG с акцентом на использовании элемента `<ellipse>`. Анализируются основные атрибуты: координаты центра, радиусы и поворот. Приводятся практические примеры преобразования растровых изображений в векторный формат.*

N.I. Ulasevich, N.A. Zhilyak

Belarusian State Technological University
Minsk, Belarus

CONVERTING BITMAP IMAGES TO SVG BASED ON THE ELLIPSE ELEMENT

***Abstract.** This article explores the possibilities of converting images to SVG, with a focus on using the `<ellipse>` element. It analyzes the main attributes, such as center coordinates, radii, and rotation. The article provides practical examples of converting raster images to vector format.*

В эпоху цифровых технологий векторная графика занимает всё более важное место благодаря своей масштабируемости и гибкости. Формат SVG (Scalable Vector Graphics) позволяет создавать высококачественные изображения, которые не теряют чёткости при любом увеличении, что делает его незаменимым инструментом для веб-дизайна, иллюстраций и анимации. Одним из ключевых элементов SVG является `<ellipse>` — мощный инструмент для создания разнообразных геометрических форм, от простых окружностей до сложных графических композиций.

Элемент <ellipse> в SVG определяется несколькими основными параметрами, которые позволяют точно настраивать его форму и положение. Ключевые атрибуты включают: координаты центра эллипса по осям X и Y, которые определяют его положение на холсте, радиусы по горизонтали и вертикали, задающие ширину и высоту эллипса и параметр, позволяющий применять повороты, масштабирование и другие преобразования для создания более сложных форм.

Благодаря этим параметрам <ellipse> становится универсальным инструментом для создания как простых геометрических фигур, так и сложных графических элементов.

Для преобразования изображения будут рассмотрены два подхода с переносом изображения в svg с применением переноса по пикселям и с применением группировок. В первом методе используется разбиение изображения на сетку с неперекрывающимися участками с интерполяцией цвета в нём. Форма двумерной области может быть любой как пример треугольной [1], прямоугольной [2] или даже любой другой неправильной формы как пример эллипса или круга. В данной реализации метода каждая двумерная область исходного изображения заменяется векторным элементом в нашем случае, <ellipse>. Такой подход может позволить максимально точно передать детали изображения, однако приводит к созданию большого количества векторных объектов, что увеличивает размер итогового SVG-файла и может замедлить его отображение. Для максимально достоверного отображения размер области может равняться одному пикселю исходного изображения. Метод прост в реализации для небольших изображений, но становится неэффективным при работе с высоким разрешением. Например, для преобразования изображения будет использован логотип представленный на рис. 1. С разрешением приблизительно 800 пикселей по ширине и высоте и размером 68 Кб.



Рис. 1 - Исходное изображение для преобразования

В данном изображении можно увидеть большое число мелких деталей, которые хорошо подходят для демонстрации преобразования.

В рамках разработанного кода имеется возможность выбора точности получаемого преобразованного сообщения. Преобразованное изображение представлено на рис. 2.



Рис. 2 - Преобразованное попиксельно изображение

В данном случае после преобразования каждого пикселя можно заметить отличия только в рамке вокруг логотипа. Возможно также уменьшения размера изображения как пример преобразования будет ограничено числом элементов на одну сторону. Для демонстрации программа была запущена с 100 элементами по ширине и высоте изображения и результат преобразования представлен на рис. 3.



Рис. 3 - Преобразованное попиксельно изображение с заданным разрешением

Полученное изображение имеет заметные искажения в отображении, что особенно сильно затронуло буквы. Результат сравнения размеров файла и числа строк представлен в таблице 1.

Таблица 1- Сравнение размеров файлов после преобразования.

Разрешение после преобразования в эллипсах	Количество строк в файле	Размер файла Кб
100x100	4275	476
200x200	16584	1843
300x300	36621	4068
400x400	65257	7248
Исходное разрешение файла	259292	28776

Можно заметить, что увеличение значений получаемого изображения разрешения в два раза приводит приблизительно к 4-х

кратному росту как размера файла, так и такому же увеличению числа строк.

При использовании в свою очередь метода на основе кластеризации с использованием алгоритма K-means. В данном подходе предполагается группировка пикселей по цветовым признакам, что позволяет значительно сократить количество векторных элементов. Алгоритм разбивает изображение на кластеры, каждый из которых представляется одним или несколькими эллипсами, параметры которых такие как цвет, размер, положение определяются средними характеристиками кластера. Такой метод обеспечивает баланс между точностью и размером файла, но требует более сложной реализации и может терять мелкие детали при недостаточном количестве кластеров, кроме того, кластеризация является затратной операцией что также сужает максимальное число кластеров для формирования изображения. На рис. 4 представлено преобразованное изображение с использованием 1000 кластеров.

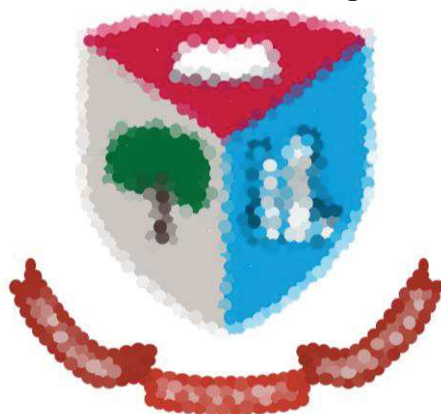


Рис. 4 - Преобразованное изображение с использование кластеризации

В данном случае все 1000 элементов были использованы для преобразования изображения. В областях не используются элементы эллипса. Кроме того, как пример отличия преобразований на рисунке 5 представлен результат преобразования с использованием 4270 кластеров что соответствует результату попиксельного преобразования размером 100 на 100 эллипсов.



Рис. 5 - Преобразованное изображение с использование 4270 кластеров

При сравнении рисунков 3 и 5 можно заметить, что во втором случае пропали черные границы у элементов и текст в обоих случаях является практически не читаемым. Кроме того, линии в случае использования кластеров менее четкие.

В статье рассмотрены два подхода преобразования растровых изображений в SVG с использованием `<ellipse>`: метод с использованием разбиения на сетку и группировки на основе K-means. Первый обеспечивает высокую точность, но увеличивает размер файла из-за большого количества элементов. Второй сокращает количество векторных объектов, балансируя между точностью и размером, однако может терять мелкие детали, но также.

Список использованных источников

1. Zhou, H., Zheng, J., Wei, L.: Representing images using curvilinear feature driven subdivision surfaces. IEEE transactions on image processing 23(8), 3268–3280 (2014)
2. Price, B., Barrett, W.: Object-based vectorization for interactive image editing. The Visual Computer 22, 661–670 (2006)