

Implementando o *Marching Squares* em HTML5 e JavaScript para navegadores

Vinicius Junqueira Schettino¹

¹Departamento de Ciência da Computação (DCC) — Universidade Federal de Juiz de Fora (UFJF)

`vinicius.schettino@ice.ufjf.br`

Resumo. *Aplicações web apresentam cada vez mais funcionalidades e interatividade com seus utilizadores. Tal advento está diretamente relacionado à crescente capacidade computacional dos navegadores, que cada vez mais executam tarefas complexas sem grande interferência do servidor para apoiar a experiência do usuário e a distribuição do processamento. Assim, este trabalho propõe a implementação da extração de contornos de figuras utilizando HTML5 e JavaScript, buscando apoiar funcionalidades de reconhecimento e processamento de imagens client-side com tecnologias modernas e preparadas para a nova geração de aplicações. Através do algoritmo marching squares, o contorno é encontrado e demonstrado para o usuário. Um benchmark simples ajuda a avaliar o algoritmo e futuras otimizações da codificação.*

1. Introdução

O *marching squares* é um algoritmo de detecção de contorno para matrizes bidimensionais,

2. Métodos e Ferramentas

. Pode ser utilizado para inúmeras funcionalidades dentro do processamento de imagens, notavelmente para reconhecimento de padrões e seleção e extração de subconjuntos [Hanisch 2004]. A técnica também pode ser estendida para atuar sobre campos tridimensionais de maneira muito próxima ao funcionamento original [Maple 2003].

Através da implementação proposta neste trabalho, é possível desenvolver funcionalidades que dependam apenas do navegador, sem consumir recursos de rede ou processamento do servidor. Esta característica pode ser útil para aplicações de tempo real como interfaces de edição de imagens ou de reconhecimento, como jogos ou utilitários simples.

3. implementação

Através do trabalho de [Spiess 2010], observou-se o funcionamento do algoritmo e adaptações necessárias para funcionamento com JavaScript. Outras implementações também foram consultadas¹ para melhor entendimento do algoritmo e separar estes conceitos das limitações de cada linguagem.

A implementação do *marching squares* foi encapsulada em uma classe (no conceito JavaScript) separada em um arquivo específico. Os métodos deste componente foram utilizados através de um script dentro da página principal do algoritmo. O resultado pode ser visto na figura 1.

¹<https://gist.github.com/mieko/3c17c41b8be9f2598c12>

Para estilização da aplicação foi utilizado o Bootstrap v4.0.0-alpha.6². Através de um contador simples de tempo, infere-se e apresenta-se o tempo necessário para execução do algoritmo.

Como calcular o contorno da figura abaixo?

A figura abaixo é gerada a partir de uma combinação aleatória de:

- Letras
- Números
- Fontes
- Cores
- Posições

Ao todo, são oito itens. Este processo é feito através do HTML5, utilizando um `canvas`. Através do algoritmo `Marching Squares` é possível descobrir as linhas de contorno de um campo escalar bidimensional - no caso, com uma representação gráfica. Para testar seu funcionamento, clique no botão abaixo.

Extrair o contorno

Tempo Necessário : 612
Veze Executadas : 0

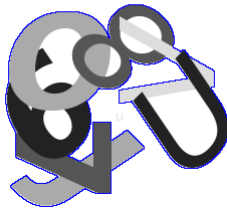


Figura 1. Funcionamento do algoritmo. O contorno azul da figura foi gerado pelo *marching squares*

4. considerações finais

O *marching squares* pode ser utilizado para diversas aplicações no âmbito de processamento de imagens, em particular na detecção e extração de padrões. Através deste trabalho apresenta-se uma implementação de tal técnica voltada para uso *client-side* em aplicações para internet, onde todo o processamento é realizado com recursos computacionais disponíveis pelo navegador.

A utilização de tecnologias modernas como o HTML5 e o JavaScript permitem a incorporação desta implementação em aplicações atuais e igualmente modernas, preparando-as para a nova geração de utilizadores e requisitos, funcionais e não funcionais.

Como trabalhos futuros, pretendemos avaliar o desempenho da implementação aqui descrita em comparação com outros métodos, levando em consideração particularidades da linguagem e do contexto ao qual se propõe esta solução.

Referências

Hanisch, F. (2004). *Marching square*.

²<https://v4-alpha.getbootstrap.com/>

- Maple, C. (2003). Geometric design and space planning using the marching squares and marching cube algorithms. In *2003 International Conference on Geometric Modeling and Graphics, 2003. Proceedings*, pages 90–95.
- Spiess, P. (2010). Better know an algorithm: Marching squares. [Online; Acessado em 12/06/2017].