Estudo do Artigo "The Art of Random Fractals"

Aluna: Giordanna De Gregoriis
Orientadora: Cristina Lúcia Dias Vaz
Co-Orientador: Dionne Cavalcante Monteiro

Universidade Federal do Pará
Instituto de Ciências Exatas e Naturais
Faculdade de Computação
Bacharelado em Ciência da Computação

4 de julho de 2016

Introdução

- Este seminário tem como objetivo mostrar os resultados do estudo feito em torno do artigo *"The Art of Random Fractals"*, escrito por Douglas Dunham e John Shier;

Introdução

• Este seminário tem como objetivo mostrar os resultados do estudo feito em torno do artigo *"The Art of Random Fractals"*, escrito por Douglas Dunham e John Shier;

• Nas próximas secções será explicado o funcionamento e a matemática do algoritmo de demonstração, cuja autoria é de John Shier e se encontra em seu site;

Introdução

- Este seminário tem como objetivo mostrar os resultados do estudo feito em torno do artigo *"The Art of Random Fractals"*, escrito por Douglas Dunham e John Shier;
- Nas próximas secções será explicado o funcionamento e a matemática do algoritmo de demonstração, cuja autoria é de John Shier e se encontra em seu site;
- Por fim serão feitas as conclusões e o resumo dos resultados obtidos.

Algoritmo

• O algoritmo criado por John Shier possui uma base matemática que permite preencher uma região espacial com uma sequência finita de formas (ou "estampas", como ele descreve) colocadas aleatoriamente e cada vez menores;

Algoritmo

• O algoritmo criado por John Shier possui uma base matemática que permite preencher uma região espacial com uma sequência finita de formas (ou "estampas", como ele descreve) colocadas aleatoriamente e cada vez menores;

• Este algoritmo pode ser usado para produzir uma variedade de padrões que são esteticamente agradáveis, que pela sua aparente autossimilaridade são chamados de padrões fractais.

Início

Inicialmente desejava-se preencher área A com estampas em posições aleatórias, progressivamente menores, sem que estas tenham intersecção entre si. Foi descoberto através de testes que isso pode ser alcançado se for obedecido uma regra de potência inversa;

Início

Inicialmente desejava-se preencher área A com estampas em posições aleatórias, progressivamente menores, sem que estas tenham intersecção entre si. Foi descoberto através de testes que isso pode ser alcançado se for obedecido uma regra de potência inversa;

Para $i = 0, 1, 2, ...$ a área da estampa de ordem i, A_i pode ser definido por:

$$A_i = \frac{A}{\zeta(c, N)(N + i)^c} \tag{1}$$

Onde $c > 1$, $c_{max} \approx 1.48$ e $N > 1$ são parâmetros, e $\zeta(c, N)$ é a função zeta de Hurwitz:

$$\zeta(c, N) = \sum_{k=0}^{\infty} \frac{1}{(N + k)^c}$$

Função Zeta de Hurtwitz

A função zeta de Hurwitz implementada no algoritmo original é feita apenas no domínio dos números reais, feita com somatórios aproximados. Esta função aproximada é dada por:

$$\zeta(c, N) = \left(\sum_{i=N}^{100000} i^{-c} \right) + \left(\frac{1}{c-1} \times N^{1-c} \right) \tag{2}$$

Função Zeta de Hurtwitz

No algoritmo de demonstração o c é um valor encontrado aleatoriamente entre 1 e 1.48 e $N = 2$.

O número retornado por esta função será utilizado para determinar a porcentagem da área A que será preenchida por A_0.

Função Zeta de Hurtwitz

No algoritmo de demonstração o c é um valor encontrado aleatoriamente entre 1 e 1.48 e $N = 2$.

O número retornado por esta função será utilizado para determinar a porcentagem da área A que será preenchida por A_0. **Exemplo:**

$$\zeta(1.263, 2) = \left(\sum_{i=2}^{100000} i^{-1.263}\right) + \left(\frac{1}{1.263 - 1} \times 2^{1-1.263}\right) \tag{3}$$

$$\approx 3.23 + 0.19 \approx 3.42$$

Início

Com $c = 1.263$ e $N = 2$, o valor obtido de $\zeta(1.263, 2)$ foi aproximadamente 3.42. O algoritmo então procede calculando a razão através de:

$$Razão = \frac{1}{\zeta(1.263, 2)} = \frac{1}{3.42} \approx 0.29 \tag{4}$$

Então a área A_0 da primeira forma a ser posicionada no plano deverá preencher aproximadamente 29% da área original.

Início

Na demonstração original deseja-se preencher o plano com círculos. Para isso utiliza-se a razão para descobrir qual será o raio do primeiro círculo que terá área A_0, dada por:

$$Raio_{A_0} = (\sqrt{A_{total}} \times \sqrt{\frac{Razão}{\pi}}) \times N^{-\frac{c}{2}} \tag{5}$$

Onde $N^{-\frac{c}{2}}$ serve para reduzir o valor obtido de acordo com a iteração N, assim servindo para a parte iterativa do algoritmo também.

Início

Substituindo a fórmula anterior pelos valores do exemplo dado alguns slides atrás e considerando que a nossa área é uma tela de 600×600 pixels, temos:

$$Raio_{A_0} = (\sqrt{360000} \times \sqrt{\frac{0.29}{\pi}}) \times 2^{-\frac{1.263}{2}} \tag{6}$$

$$\approx (600 \times 0.3) \times 2^{-0.6315} \approx 180 \times 0.64 \approx 115.2$$

Iteração

• Quando $i = 0$, a primeira forma com área de A_0 é colocada aleatóriamente no interior da região R tal que não se sobreponha a fronteira de R. Isso normalmente requer várias tentativas em posições aleatórias antes da obtenção de um posicionamento bem sucedido em que a forma é completamente dentro de R;

Iteração

• Quando $i = 0$, a primeira forma com área de A_0 é colocada aleatóriamente no interior da região R tal que não se sobreponha a fronteira de R. Isso normalmente requer várias tentativas em posições aleatórias antes da obtenção de um posicionamento bem sucedido em que a forma é completamente dentro de R;

• Quando $i > 0$, para cada $i = 1, 2, ..., n$ coloca-se de forma iterativa aleatoriamente uma cópia da estampa com área A_i dentro de R e de modo que não intersecte com qualquer cópia anteriormente colocada. Em seguida, passamos a colocar a próxima estampa, com área de A_{i+1}, ou parar o algoritmo se foi colocado a n-ésima forma ou atingiu uma outra condição de parada.

Iteração

O raio da forma A_i i para $i > 0$ é determinado por:

$$Raio_{A_i} = Raio_{A_0} \times i^{-\frac{c}{2}} \tag{7}$$

Onde $Raio_{A_0}$ é o raio da primeira forma inserida no plano.

Iteração

• Após determinar uma posição aleatória no plano, verifica-se caso a forma A_i não intersecta com nenhuma outra já posicionada. Este teste depende de cada forma, devendo ser implementado de acordo com suas características. Caso seja desejável inserir formas diferentes no mesmo plano, o teste deve ser abstraído para aceitar qualquer entrada e produzir uma resposta correta.

Iteração

• Após determinar uma posição aleatória no plano, verifica-se caso a forma A_i não intersecta com nenhuma outra já posicionada. Este teste depende de cada forma, devendo ser implementado de acordo com suas características. Caso seja desejável inserir formas diferentes no mesmo plano, o teste deve ser abstraído para aceitar qualquer entrada e produzir uma resposta correta.

• Neste algoritmo demonstratívo, o teste é feito apenas para círculos. Além disso ele verifica se a distância entre os círculos é maior que a soma de seus raios, evitando assim que os círculos sejam posicionados muito próximo entre si e tornando a imagem mais esteticamente agradável.

Iteração

Algorithm 1 Pseudocódigo do teste de intersecção dos círculos

Require: *Circulo* c2

1: $\nabla x = |c2.x - this.x|$
2: **if** $\nabla x < (c2.raio + this.raio)$ **then**
3: $\quad \nabla y = |c2.x - this.x|$
4: $\quad$ **if** $\nabla y < (c2.raio + this.raio)$ **then**
5: $\quad\quad$ **if** $this.distancia(c2) >= (c2.raio + this.raio)$ **then**
6: $\quad\quad\quad$ **return** *true*
7: $\quad\quad$ **end if**
8: $\quad$ **end if**
9: **end if**
10: **return** *false*

Iteração

Algorithm 2 Pseudocódigo que retorna a distância entre dois círculos

Require: *Circulo c2*

1: **return** $\sqrt{(this.x - c2.x)^2 + (this.x - c2.y)^2}$

Finalização

Terminado a iteração temos uma figura que se assemelha a um fractal geométrico aleatório onde nenhuma das estampas se tocam, e a área não preenchida ou carpete é um conjunto conectado contínuo (isso se forem estampas sem oco).

Finalização

Exemplo de uma execução:

Testes e Resultados

Vários testes foram realizados para entender melhor a relação entre os diversos parâmetros presentes no algoritmo.

Círculo

Círculo

Círculo

Quadrado

Quadrado

Quadrado

Conclusão

Referências