

MC930A – Computação Gráfica - 2017-S1 - Jorge Stolfi

Trabalho de laboratório 07 - 2017-05-26

Oitenta Tentáculos Estonteantes

Nome	RA	Nota
------	----	------

Objetivos. Treinar princípios de animação por computador

Enunciado. Sua tarefa nesta aula prática é contruir uma *animação do tentáculo* modelado em exercícios anteriores, usando POV-Ray. O movimento do tentáculo será obtido por interpolação de quadros-chave, como será explicado no laboratório.

Parte 1. Antes de começar a programar, desenhe no espaço abaixo (à mão livre, em perspectiva aproximada) um esboço dos quadros chave de sua animação. Indique o tempo chave (valor de `clock`) para cada quadro chave. Use pelo menos 6 quadros chave, sendo o último igual ao primeiro. **Esta parte deve ser executada nos primeiros 15 minutos de aula.** Faça o desenho a lápis, mas, **no final dos 15 minutos, passe tinta sobre o mesmo.**

Teoria: Um vídeo é basicamente uma sequência de N imagens (*quadros*, ou *frames*), que devem ser visualizadas rapidamente em sequência (tipicamente 30 por segundos). A maneira usual de criar uma animação com POV-Ray é modelar uma cena que depende de um parâmetro `clock`. Os quadros da animação podem ser então produzidos executando-se o POV-Ray N vezes, com valores diferentes de `clock`.

Cada execução produz um único quadro na forma de uma imagem `.png`. No final, esses quadros podem ser juntados para formar um único arquivo `.gif` animado (ou um vídeo em outro formato qualquer).

Uma técnica comum para contruir cenas que dependem do tempo é especificar uma lista de $m + 1$ cenas chave $CK_0, CK_1, CK_2, \dots, CK_m$, que descrevem a cena desejada em certos tempos chave $tK_0, tK_1, tK_2, \dots, tK_m$. Para construir a cena $C(tt)$ para um determinado tempo $tt = \text{clock}$, tal que $tK_0 \leq tt < tK_m$, basta então

1. determinar um índice i tal que $tK_i \leq tt < tK_{i+1}$;
2. interpolar os parâmetros de CK_i e CK_{i+1} para o tempo tt ;
3. usar esses parâmetros para construir a cena $C(tt)$.

Parte prática. Escreva e teste as macros abaixo, conforme especificado na classe:

- `busca_tempo(tt,m,tK)`: Supõe que `tK` é um array de $m + 1$ tempos chaves, em ordem não-decrescente, e que `tt` é um valor entre `tK[0]` e `tK[m]`. Devolve o (único) inteiro i tal que $tK[i] \leq tt < tK[i + 1]$.
- `define_tentaculo_quadro_chave(i,N,P,R)`: supõe que `P` é um array de pontos e `R` é um array de números, ambos com dimensões `[N]` `[4]`. Preenche esses arrays com os dados (curvas-eixo e raios) do tentáculo no quadro chave de índice $i \in \{0..m\}$.
- `define_tentaculo_quadro_geral(tt,N,P,R)`: Supõe `N,P,R` como acima, e $0 \leq tt < 1$. Preenche `P` e `R` com os parâmetros do tentáculo para o quadro genérico de tempo `tt`.

Observações. O arquivo de descrição `main.pov` deve ser construído manualmente, com um editor de texto comum, **sem** o auxílio de qualquer editor gráfico ou outra ferramenta de modelagem geométrica. Não é permitido copiar ou incluir quaisquer arquivos POV-Ray além dos fornecidos pelo professor ou escritos por você mesmo, neste exercício ou em exercícios anteriores.

Lembre-se de que todo trabalho prático é **individual**. **Não se esqueça de executar o comando `make export` até o final da aula.**