

Parte 2

Cena Simples Interativa com Câmara Móvel e Colisões

Objectivos

Os objectivos da segunda parte dos trabalhos de laboratório são explorar o conceito de câmara virtual, as diferenças entre câmara fixa e câmara móvel, as diferenças entre projecção ortogonal e projecção perspectiva, a compreensão das técnicas básicas de animação e a detecção de colisões.

A avaliação da segunda parte do trabalho será realizada na semana de **26 a 30 de Outubro** e corresponde a **5 valores** da nota do laboratório. A realização deste trabalho tem um esforço estimado de **10 horas** por elemento do grupo, distribuído por duas semanas.

Tarefas

As tarefas para a segunda parte são:

1. Definir duas câmaras adicionais tendo o cuidado de manter a câmara definida anteriormente. Deve ser possível alternar entre as três câmaras utilizando as teclas “1”, “2” e “3”. A câmara 2 deve ser fixa e permitir visualizar todo o terreno de jogo através de uma projecção perspectiva. A câmara 3 deve também utilizar uma projecção perspectiva mas é móvel. Esta deve estar colocada atrás do carro e acompanhar o seu movimento (o carro deve estar visível). **[1,5 valores]**
2. Implementar o movimento das laranjas. Este deve ser um movimento rectilíneo uniforme, mas estas devem rodar conforme se deslocam. Diferentes laranjas devem movimentar-se com velocidades diferentes e a sua velocidade vai aumentando com o tempo de jogo, i.e quanto mais tempo o utilizador estiver a jogar, mais rápido as laranjas se movem. Após sair do campo de jogo (da mesa) as laranjas devem desaparecer e reaparecer passado algum

tempo noutro local da mesa. O aparecimento das laranjas no jogo deve seguir um comportamento aleatório. [2,0 valores]

3. Detectar a colisão do carro com as laranjas, as manteigas e os limites da estrada (os *cheerios*). Estas colisões são detectadas em duas dimensões usando caixas envolventes alinhadas com os eixos (axis aligned bounding boxes - AABB), conforme exemplificado na Figura 1, e fazendo comparações dos limites das caixas, conforme ilustrado na Figura 2. Quando atingir os limites da pista (colida com um *cheerios*) ou com uma manteiga, o carro deve parar, mesmo que o utilizador mantenha premida a tecla que o movimenta para o lado onde a colisão está a ocorrer. Ao serem tocados pelo carro as manteigas ou os *cheerios* movem-se ligeiramente. Ao colidir com uma laranja o carro perde a corrida e volta à posição inicial, na linha da partida. [1,5 valores]

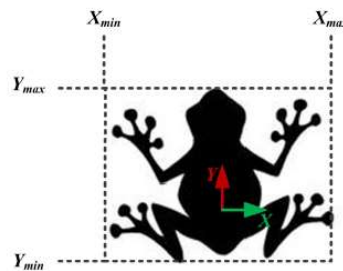


Figura 1 - Caixa envolvente alinhada com os eixos (AABB)

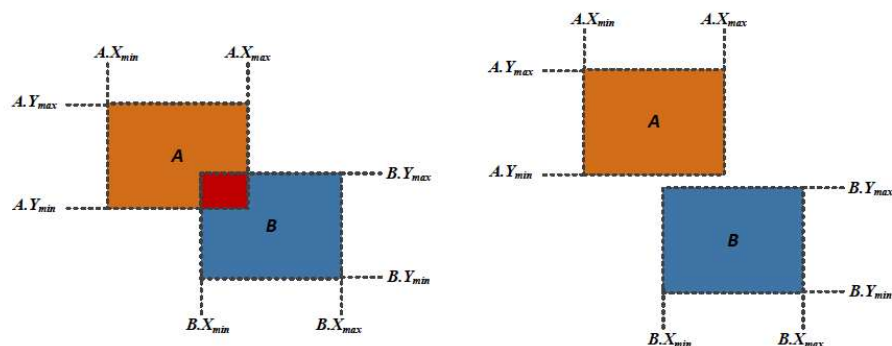
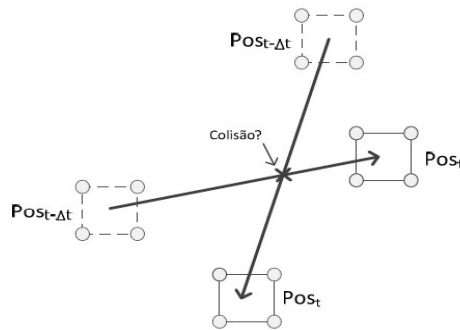


Figura 2 - Exemplo de detecção de colisão entre caixas envolventes alinhadas com os eixos (AABB) em duas dimensões. Esquerda: com colisão. Direita: sem colisão.

Sugestões

1. Deve usar um temporizador com um intervalo aleatório para definir quando uma laranja volta a aparecer depois de ter caído da mesa. A posição e velocidade com que volta a aparecer também devem ser aleatórias.
2. Para incrementar a velocidade das laranjas com o tempo de jogo é recomendado o uso de um temporizador com algumas dezenas de segundos. Ao disparar o temporizador, a velocidade das laranjas aumenta ligeiramente. Pode-se acrescentar um caule ou uma folha à laranja para se ter percepção de movimento.
3. Para obter um resultado mais correcto, a colisão entre objectos deve ser detectada considerando o percurso realizado por estes entre a posição actual (Pos_t) e a posição anterior ($Pos_{t-\Delta t}$), tal como ilustrado na figura abaixo.



4. A implementação da detecção de colisões deve ser única para todos os objectos. Esta afirmação é válida tanto para colisões entre o carro e as laranjas, as manteigas ou os limites da estrada, como para as colisões das laranjas com os limites do campo de jogo (a mesa).
5. Algumas das funções a estudar:
 - `gluPerspective`, `gluLookAt`
 - `glutTimerFunc`
 - `glutIdleFunc`
 - `glutGet(GLUT_ELAPSED_TIME)`
 - `glEnable(GL_DEPTH_TEST)`
 - `rand`, `srand`