

Primeiro Trabalho – Cena Simples e Interactiva com Câmaras Fixas



Figura 1 – Exemplo de uma cena 3D inspirada nos trabalhos de Wassily Kandinsky (Fonte: Wassily Kandinsky Animation - <https://www.youtube.com/watch?app=desktop&v=81fkBLIzvcM>). A pintura abstracta de Wassily Kandinsky não tenciona representar a realidade física e caracteriza-se pela perfusão de cores e de formas geométricas diversas, incluindo linhas rectas e curvilíneas bem como reticulados.

Objectivos

Os objectivos deste primeiro trabalho de laboratório consistem em (i) compreender e implementar a arquitectura de uma aplicação gráfica interactiva, assim como (ii) explorar os conceitos básicos de modelação geométrica por instanciação de primitivas.

Todos os grupos devem submeter o código até ao dia **20 de Maio às 23:59h**. As discussões serão realizadas nos respectivos turnos na semana de 23 a 27 de Maio. O Primeiro Trabalho corresponde a **4 valores** da nota do laboratório. A realização deste trabalho tem um esforço estimado de **10 horas** por elemento do grupo, distribuído por **duas semanas**.

Não esquecer de comunicar ao docente do laboratório as **horas despendidas pelo grupo (média do grupo)** na realização deste trabalho.

Lista de Tarefas

1. Com “papel e caneta”¹ esboçar uma composição artística de uma cena estilo ‘Kandinsky’ (Figura 1). O esboço deve apresentar (i) uma figura geral ilustrando a composição pretendida com os vários objectos, bem como (ii) outras pequenas figuras onde são definidas as dimensões que se querem atribuir aos objectos da composição. Podem encontrar mais detalhes e uma melhor descrição dos requisitos de modelação 3D na legenda da Figura 1 e no texto da Tarefa 3, pelo que as devem seguir à risca. No anexo A pode encontrar um exemplo de um esboço em papel para um objecto (Figura A.1). **[0,25 valores]**
2. Numa outra folha de “papel”¹ definir o grafo de cena do objecto articulado. De todos os objectos que serão incluídos na composição ‘Kandinsky’, devem seleccionar 3 (três) que irão articular-se entre si (i.e., apresentam uma hierarquia de transformações geométricas entre si). Neste desenho devem ficar explícitos quais são as primitivas geométricas, quais as transformações geométricas a aplicar com os seus parâmetros numéricos bem como quais as relações de parentesco entre as peças que compõem o objecto articulado. No anexo B podem encontrar um exemplo de um grafo de cena para um objecto articulado (Figura B.1). **[0,25 valores]**
3. Modelar em Three.js a composição ‘Kandinsky’. Deverão ser usados objectos geométricos disponíveis na biblioteca Three.js^{2,3} com as formas de cilindros, cubos, esferas, cones, torus, cilindros generalizados (TubeGeometry) e poliedros regulares. Cada composição deverá apresentar entre 15 a 20 instâncias destas formas por forma a emular um quadro de pintura abstracta ao estilo de Wassily Kandinsky. Note-se que as primitivas não devem estar confinadas a um plano, mas antes, devem ocupar diferentes pontos no espaço tri-dimensional da cena. **[1,0 valores]**
4. Definir uma câmara fixa com uma vista frontal sobre a cena utilizando uma projecção ortogonal que mostre toda a cena usando a **tecla ‘1’**. Definir mais duas câmaras fixas com vistas topo e lateral utilizando sempre projecções ortogonais usando as **teclas ‘2’ e ‘3’**. A representação visual destes objectos deve alternar entre modelo de arames e sólida usando a **tecla ‘4’**. **[0,5 valores]**.
5. Permitir ao utilizador movimentar o objecto articulado com o teclado utilizando as teclas ‘Q(q)’ e ‘W(w)’ para controlar o ângulo ϑ_1 que roda todo o objecto, as teclas ‘A(a)’ e ‘S(s)’ para controlar o ângulo ϑ_2 de um ramo secundário e as teclas ‘Z(z)’ e ‘X(x)’ para controlar o ângulo ϑ_3 de um ramo terciário. Permitir ainda ao utilizador deslocar o objecto articulado pelo espaço da cena com o teclado utilizando as teclas das setas e as teclas ‘D(d)’ e ‘C(c)’ para o reposicionar segundo os eixos globais dos X, Y e Z, respectivamente. Os movimentos do objecto articulado deve apresentar um movimento a velocidade constante, sendo a direcção do movimento dada por um vector tridimensional. O cálculo da movimento deve ter em consideração que o utilizador pode carregar em várias teclas em simultâneo. **[1,0 valores]**

¹ Por “papel” entenda-se optar por um material celulósico (papel analógico) ou por um dispositivo multi-toque como um tablet, smartphone, laptop 2-em-1 (papel digital). Devem ser apresentados desenhos à mão livre pelo que não devem recorrer a templates nem a desenho vectorial de formas idealizadas.

² Vários exemplos podem ser encontrados na documentação oficial <https://threejs.org/manual/#en/primitives>

³ Ilustração com várias primitivas Three.js <https://threejs.org/manual/examples/primitives.html>

Notas Importantes:

1. Antes de escrever qualquer linha de código, é necessário esboçar o que se pretende modelar em 3D pois tal actividade ajuda muito a perceber que primitivas e transformações devem ser aplicadas. Não menos importante é o desenho do grafo de cena, enquanto representação abstracta dos objectos, pois consiste num diagrama fundamental para a correcta modelação não só dos objectos mas como de toda a cena.
2. A implementação de todos os trabalhos desenvolvidos nos laboratórios de Computação Gráfica deve usar o ciclo de animação (update/display cycle). Este padrão de desenho, usado nas aplicações de computação gráfica interactiva, está ilustrado na Figura 2 e separa o desenho da cena no ecrã da actualização do estado do jogo em duas fases distintas. Na fase de display são cumpridos três passos base: limpar o buffer; desenhar a cena e forçar o processamento dos comandos. Na fase de update todos os objectos do jogo são actualizados de acordo com a física inerente. É ainda nesta fase que se processa a detecção de colisões e implementação dos respectivos comportamentos.

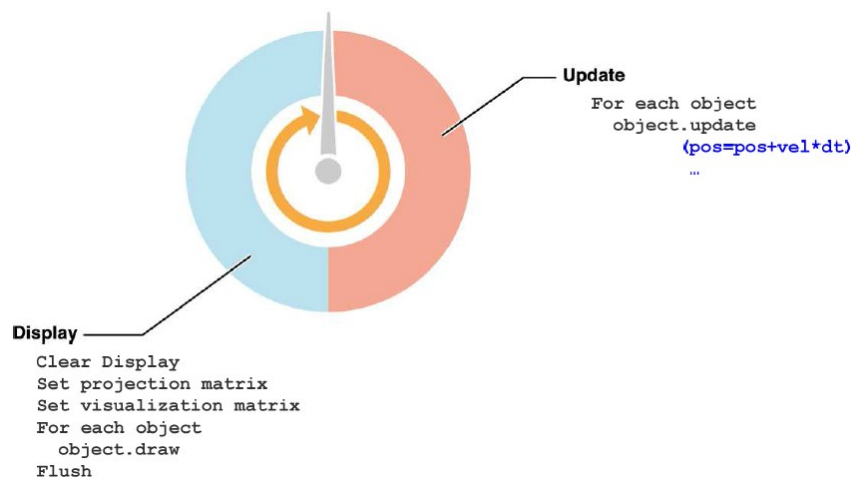


Figura 2 – Ciclo de animação com as fases de *update* e *display*.

Sugestões

1. Para esta avaliação não existem detecção de colisões entre os objectos.
2. Para além dos acontecimentos de *update* e *display* existem mais um conjunto de acontecimentos, tais como teclas pressionadas ou soltas, temporizadores e redimensionamento da janela. Sugerimos vivamente que tais acontecimentos sejam tratados pelas respectivas funções de *callback* de forma independente. **Ter em atenção que nos próximos Trabalhos iremos requerer a implementação devida dos acontecimentos de redimensionamento da janela!**
3. Por fim, os alunos devem adoptar uma programação orientada a objectos, seguindo sempre boas práticas de programação que permitam a reutilização do código em entregas posteriores e facilitem a escalabilidade.

Anexo A

Esboço de Objectos

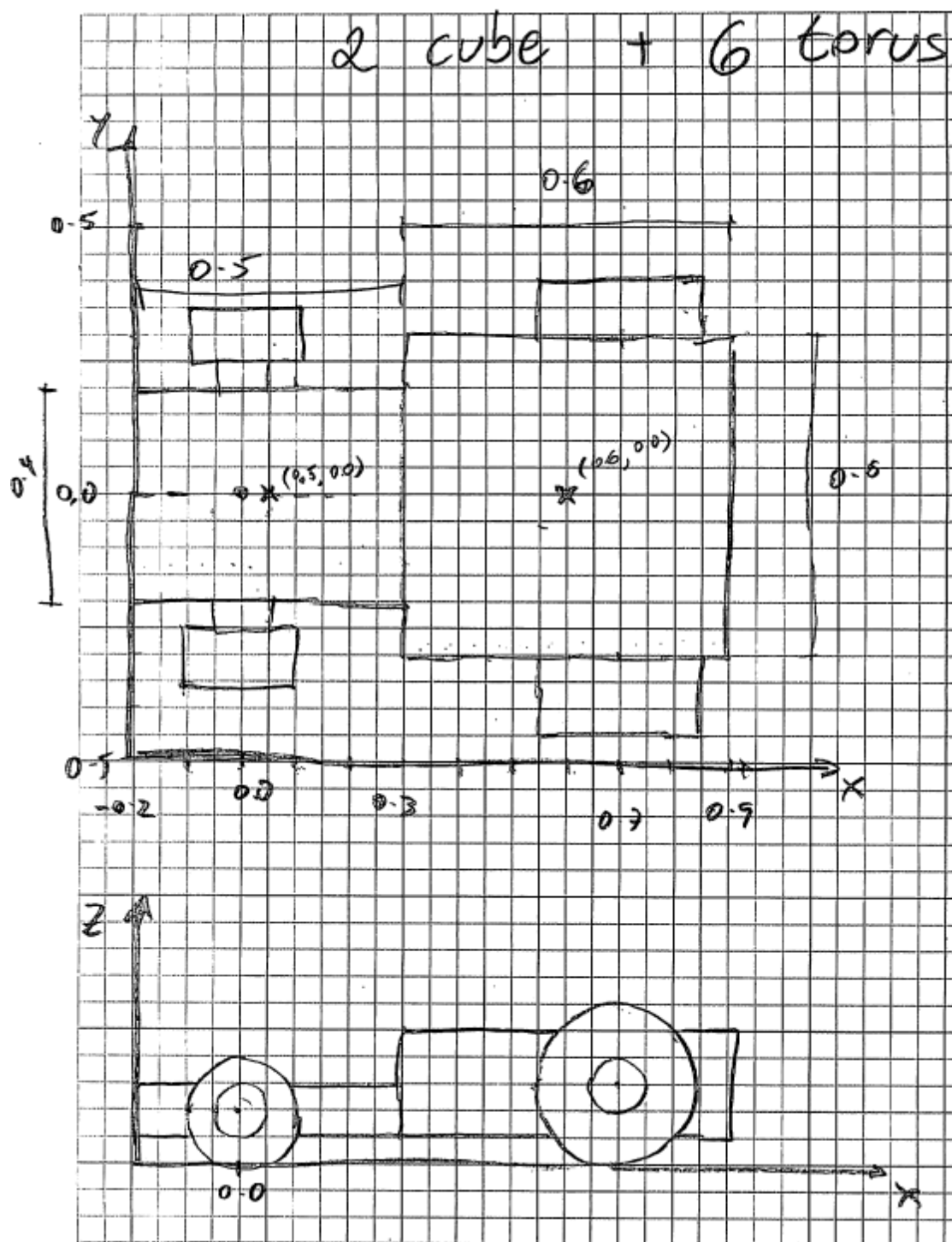


Figura A.1 – Esboço exemplificativo de um carro simples. Note-se que podem desenhar recorrendo a outras vistas ou perspectivas. O importante é que o esboço reúna as características necessárias para servir de suporte à modelação 3D.

Anexo B Grafo de Cena

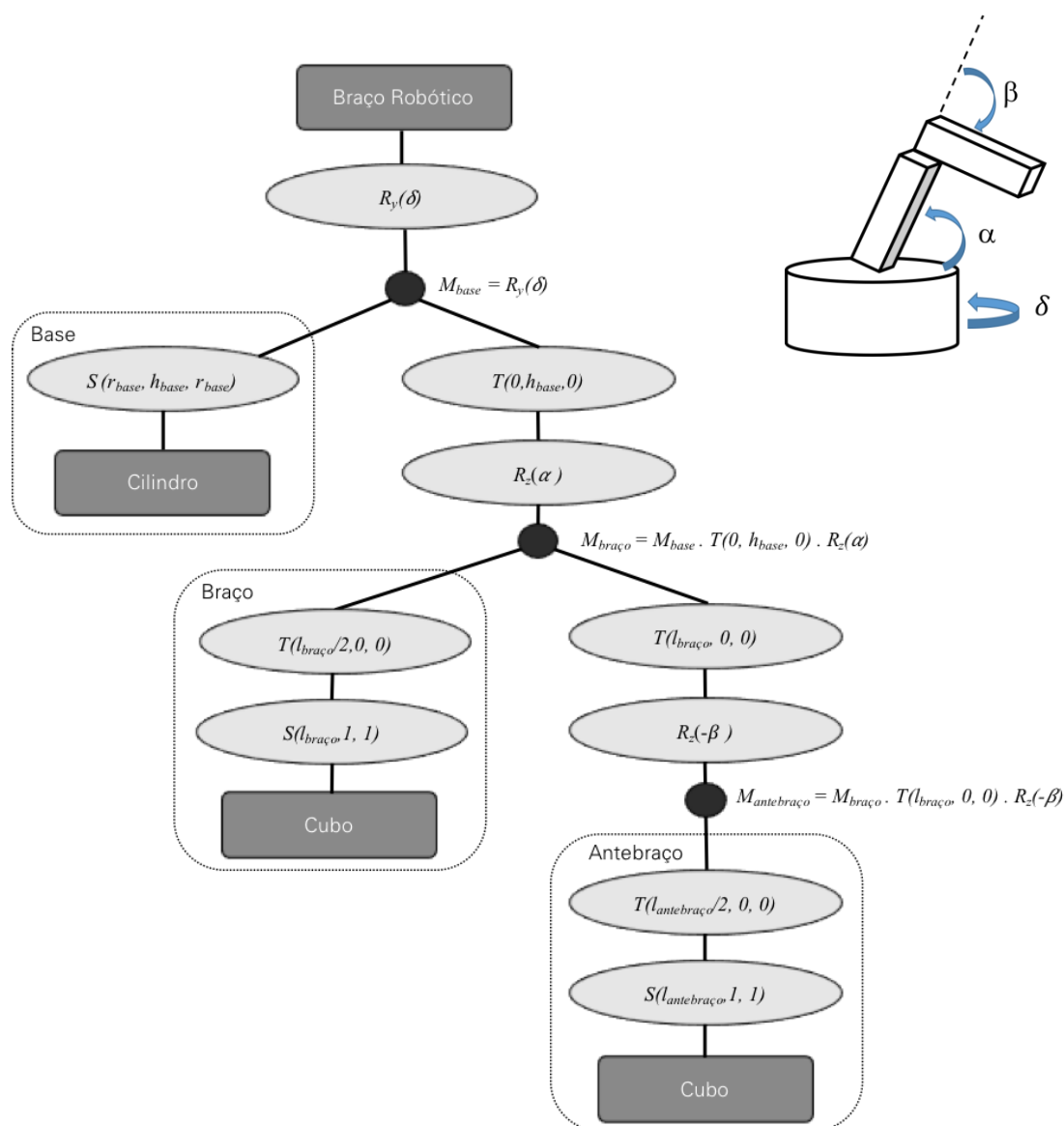


Figura B.1 - Grafo exemplificativo de um braço robótico simples ou de um pêndulo duplo (explicação proveniente do livro da cadeira).