

Terceiro Trabalho – Cena Interactiva com Malhas, Materiais, Luzes, Mensagens, Texturas e Câmara Estereoscópica



Figura 1 – Ilustração com três etapas principais na construção de um cisne origami (segundo a [origami.plus - Swan](https://origami.plus/swan))

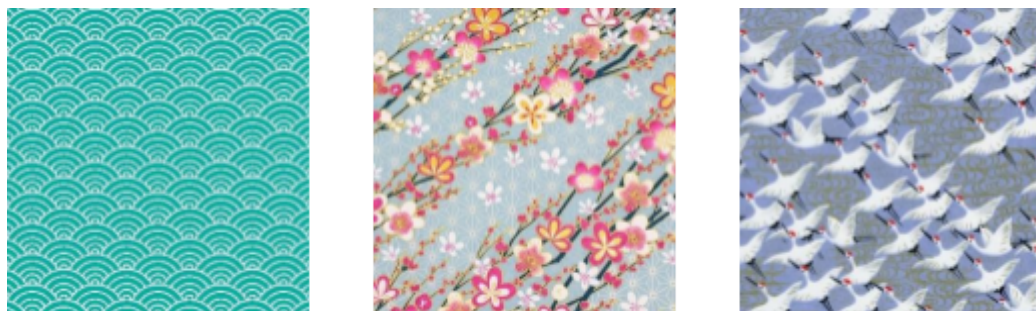


Figura 2 – Exemplos de padrões encontrados em papel origami.

Objectivos

Os objectivos do terceiro trabalho de laboratório são (i) perceber as noções básicas de iluminação, (ii) os conceitos de material, (iii) de fontes de luz direccional e *spotlight*, (iv) a modelação geométrica por instanciação de primitivas e criação de malhas de polígonos, (v) compreender os princípios básicos da aplicação de texturas, (vi) saber como implementar a reiniciação e a pausa de uma aplicação interactiva bem como (vii) os princípios fundamentais sobre a câmara estereoscópica.

Todos os grupos submetem o código até ao dia **17 de Junho às 23:59h**. As discussões serão realizadas nos respectivos turnos na semana de 20 a 24 de Junho. O Terceiro Trabalho corresponde a **3 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 a cena ilustrada na (Figura 1), composta pelas três etapas principais da construção do origami. Este esboço deve apresentar (i) uma figura geral de toda a cena ilustrando todos os elementos constituintes, bem como (ii) outras pequenas figuras onde são definidas as dimensões que se querem atribuir à última etapa da construção de um cisne origami. Para além das dimensões que se querem atribuir, no esboço deve ficar claro quais as malhas de triângulos que estão a considerar bem como as coordenadas dos vértices. Recomenda-se vivamente que construam um origami em papel (partindo de uma folha quadrada) para depois medir as dimensões e ângulos necessários para construírem os triângulos das malhas. Podem encontrar mais detalhes e uma melhor descrição dos requisitos de modelação na legenda da Figura 1 e no texto da Tarefa 2, pelo que as devem seguir à risca. **[0,25 valores]**
2. Criar uma cena que apresenta um chão plano, um palanque rectilíneo com 2 degraus. As três etapas devem ser colocadas em linha (ao longo do comprimento do palanque) e a flutuar sobre o palanque. É possível rodar, segundo o eixo vertical, as três peças origami 1, 2, e 3 recorrendo aos pares de teclas ‘Q(q)’ e ‘W(w)’, ‘E(e)’ e ‘R(r)’, ‘T(t)’ e ‘Y(y)’, respectivamente. O cálculo do movimento deve ter em consideração que o utilizador pode carregar em várias teclas em simultâneo. O chão e o palanque devem ser modelados por instanciação de primitivas. Quanto às peças origami, estas devem ser construídas recorrendo a malhas de polígonos. Por forma a facilitar a construção das malhas, sugere-se que definam apenas as faces visíveis de cada peça (portanto, não é necessário modelar as faces não visíveis!). Note-se que os vincos e as dobras devem estar visíveis (i.e., cada face tem uma normal única, portanto, não são coplanares entre si). Para simular as porções de papel correspondentes ao lado não estampado do papel origami, estas faces devem ter uma cor branca mais intensa. **[0,75 valores]**
3. Definir dois tipos de materiais (*MeshLambertMaterial* e *MeshPhongMaterial*) por cada objecto da cena. Deve ser ainda possível alternar o tipo de sombreamento entre *Gouraud (diffuse)* e *Phong* usando uma tecla (‘A(a)’), assim como, deve ser possível activar e desactivar o cálculo da iluminação usando uma tecla (‘S(s)’). **[0,25 valores]**
4. Criar a iluminação global da cena recorrendo a uma fonte de luz direcciona. A fonte de luz direcciona deve incidir com um ângulo diferente de zero relativamente à normal ao topo do palanque. Esta fonte de luz deve poder ser ligada ou desligada através da tecla (‘D(d)’). Criar um total de três holofotes (fontes de luz *spotlight*) localizados directamente sobre a superfície do palanque. Cada holofote deve apontar para uma peça origami. Esta iluminação com holofotes deve ser suficiente para se conseguir visualizar as peças e o palanque, mas não necessita de os iluminar na íntegra. As fontes de luz *spotlight* devem poder ser activadas ou desactivadas através das teclas ‘Z(z)’, ‘X(x)’ e ‘C(c)’ que ligam e desligam cada um dos holofotes individualmente. Os holofotes devem ser geometricamente modelados usando duas primitivas geométricas: um cone e uma esfera bastando atribuir um tipo de material à vossa escolha. **[0,50 valores]**
5. Usar uma imagem de papel origami (Figura 2 apresenta alguns exemplos) como textura (*colormap*) que deve ser aplicada às três peças origami. **[0,25 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 modelos nem a desenho vectorial de formas idealizadas.

6. Definir uma câmara fixa com uma vista sobre toda a cena utilizando uma projecção perspectiva que pode ser seleccionada usando a tecla '1' assim como uma câmara fixa, seleccionada usando a tecla '2', que está alinhada e centrada com o referencial do palanque e apontada por forma a visualizarem-se as três peças origami utilizando uma projecção ortogonal. Adicionar uma THREE.StereoCamera à cena para que a aplicação suporte visualização estereoscópica em dispositivos de Realidade Virtual (VR). Para tal, devem seguir a documentação oficial em como criar conteúdo VR para uma aplicação web ([How to create VR content](#)). Por forma a correr a vossa aplicação num VR browser ou no vossos smartphones, devem colocar os conteúdos do vosso projecto (i.e., index.html, sub-pasta com código JavaScript, sub-pasta com texturas) numa página online (e.g., homepage pessoal do Técnico) **[0,25 valores]**
7. Deverá permitir-se pausar a visualização quando o utilizador pressiona a tecla 'S' e retomar ao pressionar novamente a tecla. Enquanto em pausa, deve ser mostrada uma mensagem no ecrã que deverá ser sempre legível, independente da posição da câmara e das dimensões do *viewport*. Quando em pausa, deverá ser possível voltar ao estado inicial (fazer *reset* – repor o estado inicial do sistema) sem utilizar o *refresh* do navegador, premindo a tecla 'R'. Finalmente deverá fazer-se a implementação dos acontecimentos de redimensionamento da janela. Espera-se que os alunos adoptem boas práticas de programação para que seja possível reiniciar a cena (*reset*) sem ser necessário recarregar a aplicação (i.e., sem fazer o *refresh* da página). **[0,75 valores]**

Sugestões

- Antes de definirem os materiais da cena, sugerimos que comecem por fazer algumas experiências com um objecto e material simples por forma a poderem testar e perceber os vários parâmetros individualmente.
- Para obter bons resultados na iluminação de grandes superfícies, estas devem ser subdivididas em polígonos mais pequenos.
- Todas as texturas devem reagir à iluminação.
- Para a utilização de texturas em modo local é necessário configurar as permissões do navegador. O problema e a solução encontram-se descritos na documentação do three.js.

<https://threejs.org/docs/#manual/en/introduction/How-to-run-things-locally>

Alternativamente, e caso usem o Visual Studio Code como vosso editor de texto, podem instalar a extensão *Live Server* que permite criar rapidamente um servidor local:

<https://marketplace.visualstudio.com/items?itemName=ritwickdey.LiveServer>

- Para realizar a pausa basta “congelar” o tempo.
- A mensagem de pausa deve ficar sobre o jogo e pode ser conseguida através da uma aplicação de uma textura a um objecto. Não é suposto escreverem texto. Deve-se recorrer à

utilização de uma segunda projecção ortogonal e um segundo *viewport*. Esta projecção será independente da câmara activa.

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, 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.
3. Para além de 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. **Tenha em atenção que neste trabalho iremos requerer a implementação devida dos acontecimentos de redimensionamento da janela!**
4. Não podem usar ferramentas de modelação. As malhas das peças de origami devem ser modeladas manualmente, vértice a vértice.
5. 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.