

Puzzle 3D em WebGL

Daniela Simões, 76771
Cristiana Carvalho, 77682

No âmbito da disciplina de Computação Visual, foi desenvolvido um puzzle 3D, de forma a consolidar os conceitos dados sobre WebGL nas aulas. O jogo consiste na construção de figuras a partir de peças 3D.

I. INTRODUÇÃO

No âmbito da disciplina de Computação Visual, e de modo a consolidar a matéria dada, foi proposto o desenvolvimento de um projeto em WebGL. Este projeto consiste num Puzzle 3D, que foi implementado usando Bootstrap (HTML e CSS), JQuery, JSON e WebGL.

Foi desenvolvida uma interface que permite ao utilizador ter todas as funcionalidades necessárias para a execução do puzzle.

II. IMPLEMENTAÇÃO

A. MODELOS DE PEÇAS

Para cada peça foi criado um modelo num ficheiro .txt onde são armazenados as listas de vértices e as cores, como usado nas aulas. Posteriormente esses ficheiros são importados e as figuras são construídas, cada peça é um modelo individual às quais se aplicam rotações e translações de modo a ficarem numa posição final que permita ao utilizador concluir o puzzle.

B. JAVASCRIPT

Foram utilizados para o presente trabalho ficheiros javascript. De uma forma mais geral:

- *bootstrap.js*: contém o javascript responsável pelo funcionamento da interface (apenas o comportamento do HTML).
- *maths.js*: contém as muitas das funções auxiliares usadas nas aulas práticas.
- *models.js*: contém a função que permite fazer o cálculo das normais.
- *runWebGL.js*: é o responsável por inicializar o puzzle 3D e todos os eventos associados.
- *puzzle.js*: permite instanciar todas as peças que o puzzle contém, define a posição inicial de cada peça, e é o que permite fazer reset à peça para que ela volte à sua posição inicial. Define também o background do puzzle. Inicia o WebGL com o “backface culling”.
- *piece.js*: contém as características de cada peça e permite alterar as mesmas, assim como desenhá-las. Permite também que as peças sejam iluminadas quando o puzzle está completo.

C. Json

Todos os puzzles utilizados no jogo foram definidos através de um ficheiro JSON, cada ficheiro corresponde a um puzzle diferente. Cada ficheiro define cada peça pertencente ao puzzle e a sua respetiva posição inicial, final e associa-se a cada peça um nome e um ID. Este ficheiro é carregado sempre que é iniciado um novo puzzle.

III. INTERFACE DO UTILIZADOR

Para este projeto tentou-se que a interface fosse agradável e usável.

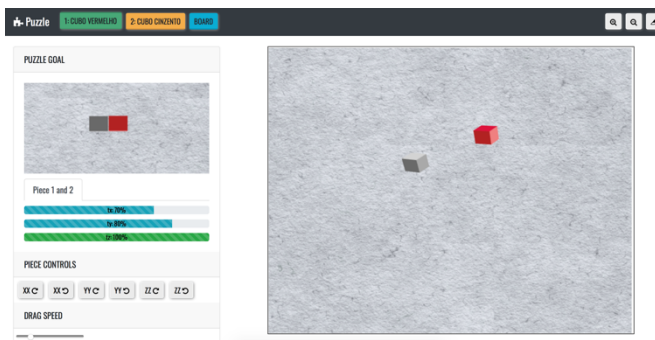


Fig. 1 - Interface Web.

A interface dispõe de comandos que permitem o utilizador fazer a rotação das peças, em XX, YY e ZZ. Apresenta a percentagem de distância que está uma peça da outra, para que seja mais fácil ao utilizador perceber se está ou não na direção certa para concluir o puzzle. Contém também a opção de “Drag Speed”; esta opção permite que seja o utilizador a escolher a velocidade mais confortável para si para trabalhar sobre o puzzle com o cursor (para “arrastar” para a direita/esquerda/cima/baixo). Existem também os controlos de zoom in, zoom out e o reset que permite que a peça selecionada volte ao seu lugar inicial. É possível mover cada peça do puzzle com

as setas do teclado ou com o cursor do rato. É ainda disponibilizada a imagem do objetivo final do puzzle. Sempre que o utilizador atingir esse objetivo o puzzle fica iluminado com uma luz verde, como mostra a figura seguinte. Para efetuar qualquer operação sobre qualquer peça, é necessário que se selecione a peça pretendida na barra de navegação.

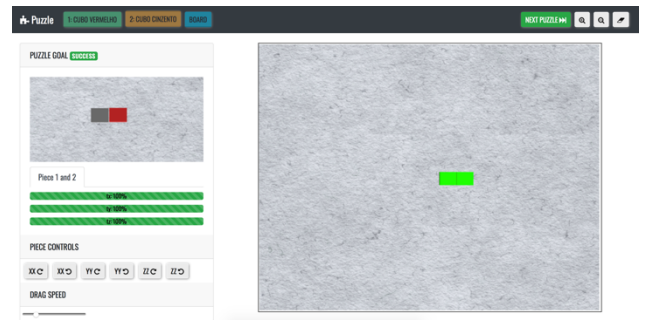


Fig. 2 - Puzzle concluído.

Quando isto acontece, é disponibilizado ao utilizador o botão para o próximo puzzle. Em cada puzzle, sempre que este é concluído com sucesso surge um som de vitória.

Como mostra a figura abaixo é possível mover a mesa e ver o puzzle de outra perspetiva.

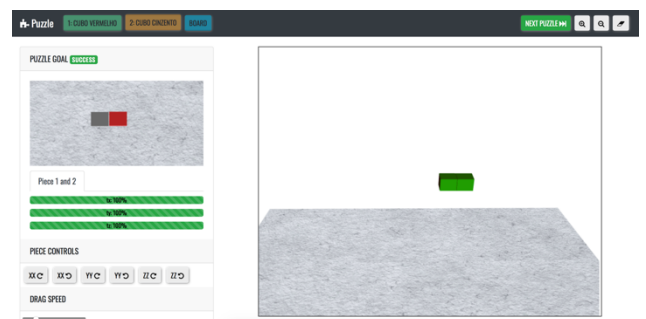


Fig. 3 - Perspectiva.

IV. INSTALAÇÃO

Para o funcionamento do Puzzle 3D apenas é necessário ter o Python instalado. Para iniciar o servidor basta:

```
# ./run.sh
```

ou

```
# python -m SimpleHTTPServer  
8080
```

posto isto, é possível aceder a:
<http://localhost:8080/>.

CONCLUSÃO

Em conclusão, foram conseguidos atingir os objetivos inicialmente propostos. Para isso foi utilizado algum código utilizado nas aulas práticas que foi, posteriormente adaptado às necessidades do projeto.

REFERÊNCIAS

- [1] Material disponibilizado nas aulas práticas.