

Modelo de descrição de experiências multissensoriais multiutilizador

Rui Rodrigues

INESC TEC e DEI/Faculdade de Engenharia da Universidade do Porto
{rui.rodrigues,eduardom}@fe.up.pt, hugomachado@gmail.com

Eduardo Magalhães

Hugo da Silva

Maximino Bessa
INESC TEC e UTAD
maxbessa@utad.pt

António Coelho
INESC TEC e DEI/Faculdade de Engenharia da Universidade do Porto
{acoelho,aas}@fe.up.pt

Resumo

Neste momento os ambientes virtuais estão a tornar-se numa peça importante nas aplicações centradas no utilizador onde se inclui o treino, design, entretenimento etc.

Apesar de percecionarmos o nosso mundo através dos nossos múltiplos sentidos, estas novas aplicações recorrem normalmente principalmente ao estímulo da visão, sendo o som o segundo estímulo mais explorado.

Descobertas recentes sugerem que quantos mais sentidos são estimulados num ambiente virtual mais imersivo é a experiência e o desempenho dos utilizadores aumenta (tal como acontece no mundo real).

O projeto MASSIVE pretende investigar em detalhe como é que podemos obter um ambiente virtual credível no qual os vários sentidos do utilizador são estimulados de forma a alcançar uma elevada imersividade.

Nesse contexto, foi desenvolvido um modelo para a descrição de experiências multissensoriais, a ser usado para representar não só as cenas, como as interações e estímulos do utilizador de forma a suportar contextos de interação multimodal e multissensorial, de forma modular. Esse modelo é apresentado neste artigo, juntamente com um exemplo prático de aplicação, demonstrando a sua relevância.

Keywords

Realidade Virtual, Experiências Multissensoriais, Modelo de descrição de dados

1. INTRODUÇÃO

O conceito de “presença” é um dos mais relevantes no contexto da realidade virtual. A sensação de presença é muito dependente do estímulo de vários sentidos, tal como acontece na vida real. Embora a evolução dos computadores, placas gráficas e o aumento da oferta de HMDs (*Head Mounted Displays*) a preços acessíveis e com qualidade aceitável tenham permitido aumentar os níveis de imersão, estes níveis continuam longe do necessário para convencer o utilizador de que realmente está “presente” num ambiente virtual. É sabido que a integração de estímulos de múltiplos sentidos, além dos mais comumente explorados - a visão e a audição - permite um salto qualitativo na sensação de “presença”.

Este tipo de ambientes virtuais multissensoriais tem sido por isso alvo de crescente interesse a nível de investigação. A sua aplicação em diferentes contextos, como o entretenimento, o treino e a certificação, o turismo e o mercado de experiências torna-os também atrativos comercialmente. No entanto, a sua implementação prática é ainda muito limitada e pouco acessível, dados os recursos e dispositivos envolvidos, e a dificuldade em criar conte-

údos para os mesmos que contemplem, por exemplo estímulos hápticos e olfativos.

Se a exploração destes ambientes por um único utilizador por si só é bastante atrativa, a possibilidade de integrar vários utilizadores em simultâneo apresenta-se como mais interessante ainda. A colaboração/competição suportada pelas redes sociais tornou-se numa mecânica comum em grande parte dos jogos. Este conceito social multi-utilizador vem sendo aplicado em ambientes virtuais como o *Second Life*¹ ou o *There*² e mantém - ou até aumenta - a sua relevância no que diz respeito a ambientes virtuais multissensoriais. O lançamento do *Oculus Rift DK2 Project Viewer*³ pela *Linden Lab* confirma o interesse das empresas responsáveis por ambientes virtuais multiutilizador na imersão proporcionada pela realidade virtual.

¹ <http://secondlife.com/>

² <http://www.there.com/>

³ <https://community.secondlife.com/t5/Featured-News/Oculus-Rift-DK2-Project-Viewer-Now-Available/ba-p/2843450>

Deste interesse na criação de ambientes virtuais multissensoriais multiutilizador, surge não só a necessidade de sistemas que implementem essas experiências, integrando os vários dispositivos de entrada e saída responsáveis pela interação multimodal e os estímulos multissensoriais associados, mas também a de mecanismos para descrever, criar, e armazenar este tipo de ambientes. São necessárias estruturas e modelos de dados, assim como ferramentas de autoria dedicadas a este tipo de experiências.

Estas lacunas, ainda por preencher, contribuem para a relevância da criação de modelos para cenários multissensoriais, como identificado por diversos autores [1,2,3,4] (ver secção 2). Em particular, é importante que um tal modelo permita definir características e elementos de uma experiência deste género a um alto nível, mas que tenha a flexibilidade para suportar um ou mais mapeamentos de diversos dispositivos, num contexto multiutilizador.

É nesse sentido que apresentamos neste artigo um modelo de representação de experiências para suporte ao desenvolvimento rápido de ambientes virtuais multissensoriais, independente do *hardware*. Um modelo deste tipo poderá permitir a geração de cenários virtuais multissensoriais e multiutilizador de uma forma relativamente expedita e disponível a utilizadores, permitindo por um lado construir bases de dados de cenários com iterações permanentes (adequando os dispositivos de controle e os conteúdos) e por outro, testar a robustez do sistema na execução desta tarefa por utilizadores “comuns”. Em conjugação com a ferramenta de autoria pode permitir a ligação direta entre a arquitetura física (motor multissensorial) e a conceptualização de experiências virtuais através da observação em tempo quase real dos objectivos pretendidos.

Esta abordagem partilha alguns dos conceitos utilizados nas arquiteturas de motor de jogo baseadas em componentes e *scripting*, como por exemplo o Kismet ou Blue Print do Unreal Engine, mas vai mais longe ao prever à partida componentes multissensoriais abstraídos, e a sua interoperabilidade com os sensores/dispositivos, a cena e os utilizadores.

Para testar a proposta, uma das provas de conceito do projeto MASSIVE é implementada segundo o modelo proposto. A experiência consiste numa descida de bicicleta por uma montanha, na qual o utilizador tem a percepção visual do ambiente campestre que o rodeia, ouvindo também os sons da paisagem sonora local, assim como pássaros ou os ruídos localizados da própria bicicleta. O aroma da montanha é também fornecido ao utilizador, assim como a trepidação do veículo ao deslocar-se sobre o solo e aquando de eventuais saltos. Em termos hápticos, é possível também sentir no peito o impacto com árvores e arbustos e o próprio vento natural da paisagem.

Este documento está organizado da seguinte forma: O trabalho relacionado é discutido na secção 2. A secção 3 descreve o conceito, requisitos e arquitetura do modelo de descrição de experiência proposto. A secção 4 está dedicada a uma proposta de descrição e armazenamento

de dados em formato XML para experiências multissensoriais e a secção 5 apresenta uma prova de conceito que consiste numa experiência de realidade virtual. A secção 6 conclui o artigo e inclui o trabalho futuro que se pretende realizar.

2. TRABALHO RELACIONADO

A necessidade de representar formalmente experiências multissensoriais tem vindo a ser discutida por diversos autores, apontando linhas de investigação e modelos semânticos diversos [2,3,4]. A vantagem destes modelos consiste na possibilidade de descrever relações multimodais entre objetos e entre utilizadores de uma forma adaptativa, proporcionando suporte a ferramentas de autoria versáteis e modulares. Desta forma é também possível uma abstração dos dispositivos que irão gerar os estímulos. Isto por um lado possibilita representar a mesma experiência utilizando conjuntos de dispositivos diferentes, mas por outro lado implica a adaptação automática do conteúdo, tendo em conta informação sobre os elementos multissensoriais de um cenário virtual, como por exemplo o número de fontes sonoras ou o tipo de *output* visual, e a inferência do seu mapeamento aos dispositivos necessários.

Gutierrez et. al [4] propõem um modelo adaptativo para Interfaces Multimodais (IMM), suportado na importância da adaptabilidade defendida por Xiao [5], que consiste em subdividir e sistematizar os cenários virtuais em duas entidades: dispositivos de interação (interfaces multimodais) e entidades virtuais (componentes 3D, imagens, texto, etc.). O modelo semântico proposto por estes autores é baseado em descritores que definem a interação entre dispositivos e mundos virtuais que, através de um sistema de canais de I/O, possibilitam que os objetos virtuais funcionem como sistemas reativos. Estes objetos poderão ter representações diversas tais como imagens, formas 2D e 3D, texto, vídeo, etc., prevendo funcionalidades “personalizáveis” ou escaláveis a novos contextos. A formalização do modelo está relacionada com a necessidade de interpretação do modelo tanto por humanos como por máquinas, pelo que propõe uma representação baseada em XML como suporte. Apesar de ser um modelo flexível, está focado essencialmente nos dispositivos de entrada e no seu mapeamento aos objetos, e não contempla de forma explícita a vertente da geração de estímulos multissensoriais

Num nível de abstração mais elevado, Irawati et al [2] descrevem uma *framework* baseada numa ontologia dos objetos que estabelece limites e regras de interação e relacionamento entre os objetos de um cenário virtual. Um dos principais objetivos é o de resolver ambiguidades relacionadas com a intermodalidade sensorial durante a manipulação de objetos 3D. Para tal propõem um modelo de descrição de cena que inclui não só as regras de interdependência entre objetos mas também as características espaciais destes mesmos objetos em relação a outros elementos que compõem o espaço. Um exemplo deste tipo de ambiguidades seria o utilizador, num dado cenário virtual, querer colocar um objecto “quadro” no objecto “parede” do cenário mas ter o objecto “homem” entre a

sua localização e o ponto para onde está a apontar. Se a relação entre o objecto “quadro” e o objecto “parede” não estiver descrita na ontologia, o sistema pode compreender que o objecto “quadro” é para ser colocado no objecto “homem” e não no objecto “parede”. Adicionalmente, o sistema permite também fazer integração de diferentes dispositivos de entrada, formando uma interface multimodal. No entanto, o mapeamento destes dispositivos não é flexível, não sendo contemplado na ontologia de objectos. Esta por sua vez contempla apenas informação espacial/tridimensional, que influencia apenas a componente visual, não incluindo outro tipo de estímulos.

Ho et al [3] descrevem uma ontologia baseada no RoSE (Representation of Sensory Effects) [9] que define as relações entre efeitos sensoriais e os dispositivos que produzem os estímulos, usando regras de inferência. Apresentam também uma aplicação - Ontology-based Multi-Sensory Inference Engine (OMSIE) - que tenta inferir estímulos a partir da informação dos efeitos sensoriais pretendidos presentes numa cena. Esta proposta é interessante no sentido de tentar automatizar a geração de estímulos através de regras. No entanto, não contempla a componente de interação multi-utilizador.

A própria proposta de standard MPEG-V inclui, na sua parte 3 [6], a proposta de uma linguagem baseada em XML para a descrição de efeitos sensoriais tais como luz, vento, nevoeiro, vibração, etc. - Sensory Effect Description Language (SEDL) [7]. Os efeitos propriamente ditos não fazem parte da SEDL, mas são definidos num vocabulário - Sensory Effect Vocabulary (SEV) - que permite abstrai-los de forma a serem instanciados em cada domínio específico, e poderem ser mapeados aos dispositivos propriamente ditos (ventoinhas, luzes, etc.). [8] apresenta um proposta de implementação, focada na extensão de sequências de vídeo/áudio com outros efeitos sensoriais. Tanto no MPEG-V como nesta implementação, o foco é na descrição dos efeitos, não contemplando propriamente a descrição de cenas e o mapeamento de efeitos a diferentes elementos de uma cena, ou a componente de interação.

Em suma, constata-se que apesar de haver por um lado, esforços no sentido de definir ontologias, modelos de dados ou formatos de descrição de cenas em ambientes interativos, e por outro de definir mecanismos de abstração de efeitos sensoriais, há ainda muito em aberto no que diz respeito à descrição integrada de cenas interativas que contemplem estímulos multissensoriais, e multiutilizador.

3. MODELO DE DADOS

Na secção anterior é mencionada a importância da representação de relações entre dispositivos de entrada e mecanismos de interação, a descrição de uma cena contemplando estímulos e o seu mapeamento a dispositivos de saída e suas características, bem como a relação com vários utilizadores.

Este é o ponto de partida para o modelo de descrição de experiência proposto neste artigo que se pretende sirva como forma de descrição de experiências multissensoriais interativas, assim como de suporte a ferramentas de

autoria deste tipo de conteúdos. O *workflow* esperado é o seguinte: alguém que pretenda criar uma experiência multissensorial usa uma ferramenta de autoria que gera um ficheiro descrevendo a experiência segundo o modelo aqui proposto; esse ficheiro é carregado por um sistema multissensorial, ao qual estarão ligado(s) o(s) utilizador(es); o sistema irá interpretar as interações dos utilizadores e gerar os estímulos adequados, de acordo com a descrição da experiência e os dispositivos de estímulos que tiver disponíveis.

O modelo prevê uma arquitetura modular por cinco tipos de entidade para descrever experiências multissensoriais: *Experiment*, *Scene*, *User*, *Inputs* e *Renderers* (figura 1). Em particular, é explorado o padrão de arquitetura baseada em componentes, de forma a possibilitar a integração de diferentes fontes de estímulos a diferentes elementos da cena.

A experiência (*Experiment*) é o bloco nuclear que agrega os restantes componentes e pode ser constituída por várias cenas. Uma cena (*Scene*) será o equivalente a um espaço/tempo, sendo que, no contexto de uma mesma experiência, um utilizador pode passar por várias cenas, mas que num dado instante apenas poderá estar presente numa cena. Um utilizador (*User*) é simultaneamente:

- um ator na cena com uma representação a vários níveis (visual, sonora, etc.),
- controlado por um utilizador real através de diferentes canais/dispositivos de entrada (*Inputs*)
- passível de influenciar/interagir com outros elementos da cena
- um recetor dos estímulos multissensoriais resultantes da experiência, gerados através de geradores desses estímulos (*Renderers*) que em última instância controlarão dispositivos de saída (HMD's, altifalantes, ventoinhas, etc.).

Estará, portanto, geralmente representado por um objecto na cena de forma a permitir a sua visualização e localização no espaço (por exemplo por outros utilizadores).

Cada uma das entidades é descrita mais detalhadamente nas subsecções seguintes.

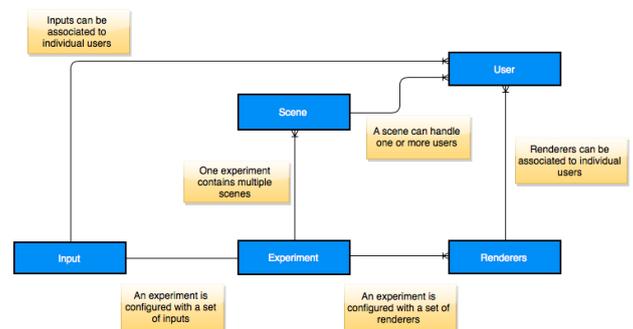


Figura 1 – Diagrama.

3.1 Scene

A cena recebe um conjunto de características de ambiente do qual constam propriedades como o local (que poderá ser baseado no mundo real ou num conceito puramente

sintético), momento temporal, condições ambientais ou os próprios valores que ditam a gravidade, entre outras propriedades relevantes.

Uma cena pode conter objetos e utilizadores, sendo que um utilizador pode estar associado a um objecto (figura 2). Árvores, veículos, animais ou mesmo humanos geridos por inteligência artificial são considerados objetos. Como tal, estes terão influência na experiência sendo visualizados, emitindo sons, cheiros ou afetando fisicamente outros objetos e utilizadores. Para tal, um objeto compreende sempre uma transformação que indica a sua posição, rotação e escala no espaço. Os objetos podem ser “pais” de outros objetos, permitindo assim situações como o exemplo em que um objeto “cesta” contenha um ou vários objetos “pão”, constituindo assim a este nível um típico grafo de cena.

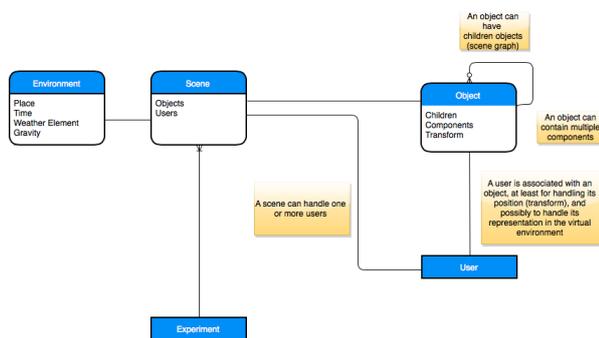


Figura 2 - Sistema relacional de Cena.

Um dos factores distintivos desta proposta é a sua arquitetura baseada em componentes. Qualquer objeto pode conter um conjunto de componentes de diferentes tipos, que definem características como as suas propriedades físicas, aroma associado, ou outras. Estas propriedades serão posteriormente utilizadas no cálculo dos impulsos fornecidos ao utilizador pelos renderizadores que lhe estejam aplicados. Com isto consegue-se uma abstração dos efeitos sensoriais, ao estarem associados aos elementos da cena, ao invés de serem globalmente declarados e despoletados. Por exemplo, uma chávena de café próxima de um utilizador dará origem à emissão do cheiro de café com uma intensidade inversamente proporcional à distância entre ambos.

O modelo contempla um conjunto de componentes que cobrem diferentes tipos de estímulos (ligados à componente visual, auditiva, háptica e olfativa). No entanto, além destes, novas ações e comportamentos podem ser adicionados por um componente de *scripting*, que poderá também permitir a recepção de dados de *input*, a interação com outros objetos, ou outros comportamentos mais específicos (figura 3).

3.2 Users

Como referido anteriormente, um utilizador no modelo proposto é um ator representado na cena, controlado por um utilizador real e que pode interagir com a cena, e receber estímulos da mesma, resultantes da experiência. Estará, portanto, geralmente ligado a um objeto de forma a permitir a sua visualização e localização no espaço - avatar. Múltiplos utilizadores podem coexistir dentro da

mesma experiência. Isto permite experiências colaborativas ou competitivas, em que, por exemplo, um utilizador poderá estar numa bancada a ver um jogo de futebol, enquanto outros estarão a experimentar as sensações de jogar o jogo no próprio campo.

Os estímulos que são fornecidos a um dado utilizador serão gerados em tempo real, tendo em conta as fontes e influenciadores virtuais de estímulos (p.ex. uma chávena de café, a relva do campo, o vento), a sua relação espacial com o utilizador em causa (distância, orientação, etc.), e os dispositivos de estímulos disponíveis (HMD, máquinas de cheiros, ventoinhas, etc.), entre outros fatores.

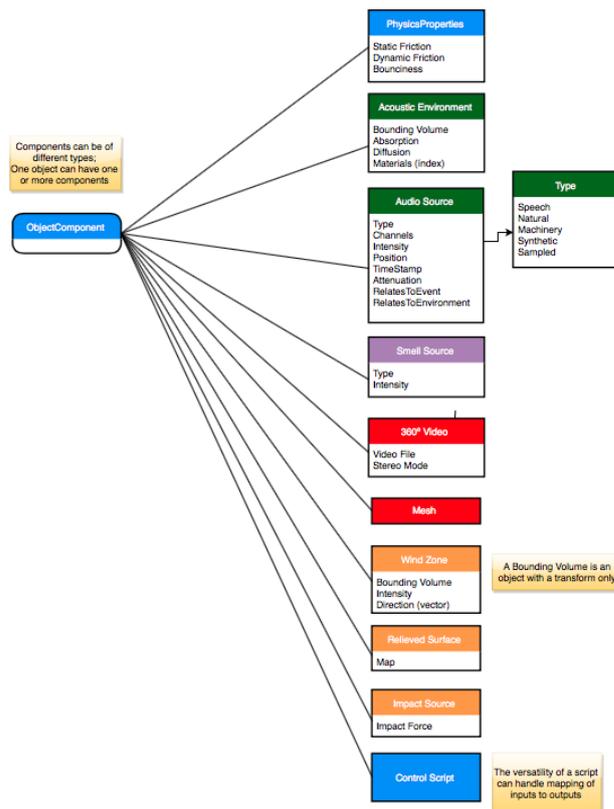


Figura 3 - Object Component e Script Component.

Esse mapeamento e geração são geridos pelos renderizadores que, numa mesma experiência, podem ser diferentes para diferentes utilizadores (p.ex., um utilizador associado a um jogador no campo pode usar um HMD para a componente visual e um colete para receber estímulos hápticos, enquanto outro utilizador associado a um espectador pode usar um monitor e não ter estímulos hápticos).

3.3 Inputs

As entradas (*inputs*) representam canais de input de médio/alto nível através dos quais um utilizador real pode controlar o seu avatar, mas que também podem ser usados para controlar outros elementos da cena (p.ex. sensor que determine a temperatura da sala real em que o utilizador se encontra). Por essa razão, as entradas podem ser diretamente associadas a utilizadores ou à própria experiência (e consequentemente a cenas individuais).

Estas entradas podem ser alimentadas por dispositivos HID (*Human Interface Device*), sensores de baixo nível,

deteção de áudio, dispositivos de *tracking* ou mesmo sensores de dados biométricos (Fig. 4). O mapeamento entre parâmetros de dispositivos de entrada e objetos ou a própria experiência simplifica o controlo, permitindo que de uma forma direta sejam conseguidos efeitos diretos (como por exemplo a translação de um utilizador aquando da pressão de uma tecla) ou indiretos, em que um sensor LDR (*Light Dependent Resistor*) permita por exemplo utilizar a luz ambiente do espaço real para controlar o ciclo dia-noite.

Mapeamentos mais diretos, como o *tracking* do utilizador real para a orientação do seu avatar podem ser representadas implicitamente mapeando o *Input* ao Utilizador. Outros mapeamentos podem ser representados recorrendo à utilização de um componente de *scripting* (descrito na secção 4) num objeto, permitindo que este seja controlado ou parametrizado por dispositivos de entrada de dados.

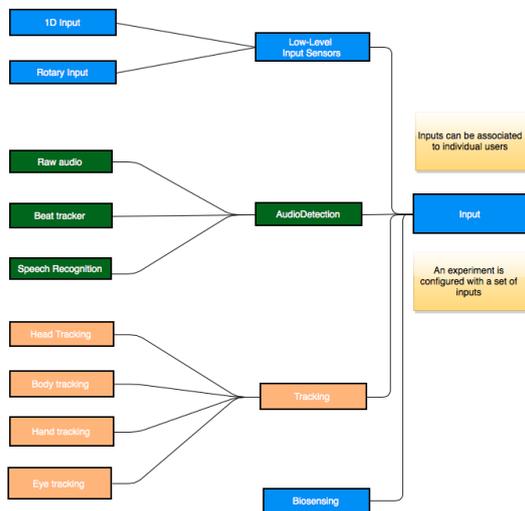


Figura 4 - Sistema de entradas através de dispositivos HID.

3.4 Renderers

Os renderizadores (*Renderers*) são as entidades responsáveis pela geração dos estímulos dependentes do utilizador, sendo portanto uma abstração dos dispositivos de saída/estímulos. Com base nos fatores ambientais, geradores de calor/frio, som, impacto e demais impulsos sensoriais, um renderizador é responsável por calcular o efeito destes fatores sobre o utilizador dependendo das condições do seu avatar e, de uma forma independente do hardware, providenciar os impulsos sensoriais ao utilizador. Uma classe de renderizadores foi identificada para cada um dos cinco sentidos e, para cada sentido, foram identificadas subclasses destes para diferentes tipos de dispositivos. Por exemplo, para a visão, pode ser associado um renderizador de uma única vista (“mono”, para visualização num monitor normal) a um dado utilizador, e um renderizador stereo (para um HMD 3D) a outro utilizador (Figura 5).

4. FORMATO DE FICHEIRO

Com base no modelo proposto, foi delineado um *schema* de XML, que permite descrever uma experiência com

base nos parâmetros estabelecidos. Este formato permite a utilização transparente entre diferentes sistemas e apresenta a vantagem de ser perceptível pela máquina, mas também legível por humanos.

Como raiz utilizamos o elemento <Experiment>, que representa toda a experiência e utiliza o atributo name para a identificar. Mais concretamente, tem como subelementos (descendentes diretos) os três elementos principais <renderers>, <inputs> e <scenes>, e um quarto elemento <controlScripts> para definir comportamentos adicionais via *scripting* (figura 7).

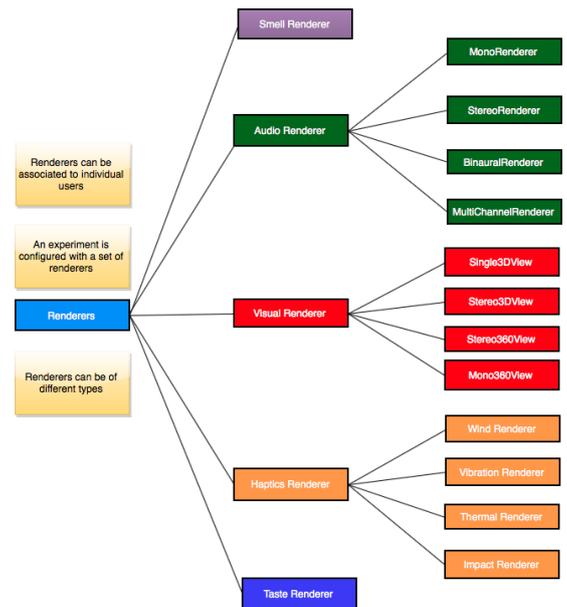


Figura 5 - Diferentes tipos de renderers.

No nó <renderers> são instanciados os renderizadores a utilizar na experiência (figura 6, secção A). Cada renderizador é definido num nó que o identifica, acompanhado de um identificador. Poderá existir também um atributo que define uma subclasse específica com informação suplementar, caso seja relevante para o renderizador.

Em <inputs> (figura 6, secção B) são descritos os dispositivos de entrada de dados utilizados. Uma vez mais, um atributo id identifica o dispositivo, e cada sub-nó poderá conter atributos específicos, caso tal seja necessário.

No elemento <Scenes> são descritas as várias cenas que compõe a experiência (figura 6, secção C). Cada cena contém utilizadores, objetos, ambientes, mas também referências aos renderizadores, inputs e demais elementos definidos das secções relevantes. A enumeração e identificação dos elementos necessários para a constituição de uma cena recorre a vários nós que descrevem o seu conteúdo.

Os utilizadores são descritos no elemento <Users>, com um subelemento <User> para cada um com o id que os identifica (figura 6, secção C.1). Caso um utilizador tenha um objeto associado, uma referência para esse objeto é indicada no atributo objectRef. Os renderizadores associados ao utilizador são referenciados na sua própria secção

<renderers>, com o elemento <RenderRef> a indicar no seu id o renderizador a utilizar.

Os *inputs* a que responde cada utilizador são da mesma forma indicados na sua secção <inputs>, sendo que cada um é referenciado pelo elemento <InputRef> que inclui a indicação do id referente à entrada relevante.

A secção *objects* (figura 6, secção C.2) contém a informação dos vários objetos presentes na cena. Cada objeto é identificado pelo elemento <object> que inclui como atributo o seu id. A posição, rotação e escala iniciais da sua transformação são definidas pelo elemento <Transform> com os atributos Position, Rotation e Scale utilizados para a indicação dos valores relevantes. A estrutura hierárquica é também fornecida no elemento object, com o(s) objeto(s) filho(s) a aparecer na secção <children>. Componentes podem também ser aqui referenciados com o elemento <ObjectComponent> que inclui a sua classe no atributo *class*. <ObjectComponent> pode também conter nós para definir scripts no nó <Script>. Subelementos do nó <ObjectComponent> podem também conter parâmetros incluídos nos atributos do elemento <param>. <MeshComponent> será o elemento utilizado sempre que um objeto tenha uma *mesh* associada.

Scripts referenciados em outras áreas do ficheiro XML podem ser definidos na secção ControlScripts (figura 6, secção D). Para isso, o elemento <ControlScript> é identificado por um id utilizado no atributo scriptRef do elemento <Script> que o utiliza. Parâmetros como valores por omissão podem ser definidos no subelemento <param>. O código pode ser definido dentro dos delimitadores de um elemento <Script>. Alternativamente, o elemento <ImportedScripts> é utilizado para importar um segmento de código de um ficheiro definido no atributo file.

5. APLICAÇÃO

No projeto MASSIVE foi desenvolvida uma série de provas de conceito, uma das quais intitulada de BikeDemo - uma descida virtual de uma montanha em bicicleta. A título de exemplo de aplicação do modelo proposto, apresentamos nesta secção um mapeamento entre os diferentes elementos desta demo - dispositivos de entrada e saída, e cena e seus elementos virtuais - com as entidades definidas no modelo. As ferramentas de autoria propriamente ditas estão ainda em desenvolvimento, pelo que parte dos componentes não é ainda importável/exportável no formato de ficheiro proposto, mas a estrutura da experiência segue o modelo proposto.

Esta experiência é bastante abrangente, tanto a nível de conteúdos, como de sensações oferecidas e mesmo de mecanismos de entrada de dados e controlo. A experiência é completamente virtual e recorre a uma bicicleta real devidamente instrumentada para permitir o controlo e movimentação por parte do utilizador (Figura 7). A velocidade é imposta pela gravidade (pedais não instrumentados), mas é controlada pelos travões; a trajetória é definida pela posição do guiador. Para a instrumentação dos travões optou-se pela utilização de um sensor de flexão, que varia a resistência conforme a dobra criada sobre ele.

```
<Experiment name="Sample">
  <!-- SECTION A -->
  <renderers>
    <SmellRenderer id="SensoryCol" />
    <VisualRenderer id="Rift1" class="Stereo3DView" />
    <AudioRenderer id="Binaural1" />
    <HapticsRenderer id="ButtKicker1" />
    <HapticsRenderer id="Vest1" />
  </renderers>

  <!-- SECTION B -->
  <inputs>
    <AudioDetection id="mic1" />
    <Tracker id="TrackIR1" class="PointTracker" />
    <LowLevelSensor id="Brake1" class="IDInput" />
    <Tracker id="Rift1" class="HeadTracker" />
  </inputs>

  <!-- SECTION C -->
  <Scenes>
    <Scene id="forest">
      <Environment
        place="Mountains"
        time="Afternoon"
        weather="Rainy"
        gravity="normal"/>

      <!-- SECTION C.1 -->
      <Users>
        <User id="Biker" objectRef="BikerObj">
          <renderers>
            <RenderRef id="Rift1" />
            <RenderRef id="Binaural1" />
          </renderers>
          <inputs>
            <InputRef id="TrackIR1" />
            <InputRef id="Rift1" />
          </inputs>
        </User>
      </Users>

      <!-- SECTION C.2 -->
      <objects>
        <object id="bike">
          <Transform
            Position="10 20 30"
            Rotation="0 0 0"
            Scale="1 1 1"/>
          <components>
            <ObjectComponent id="BikeControl"
              class="ControlScript" >
              <param name="SteeringInput" type="Input"
                value="TrackIR1" />
              <param name="BrakeInput" type="Input"
                value="FlexSensor1" />
              <Script>...</Script>
            </ObjectComponent>
          </components>
          <children>
            <object id="BikerObj">
              ...
              <components>
                <MeshComponent file="Biker.obj" />
                ...
              </components>
            </object>
          </children>
        </object>
      </objects>
    </Scene>
  </Scenes>

  <!-- SECTION D -->
  <controlScripts>
    <ControlScript id="RiftTrackingScript">
      <param name="HeadTrackingInput"
        defaultValue="null" />
      <Script> ... </Script>
    </ControlScript>
    <ImportedScripts file="LibraryScript1.xml" />
  </controlScripts>
</Experiment>
```

Figura 6 – Exemplo de ficheiro XML.

Uma placa Arduino mede os valores da resistência e envia-os pela porta série para o computador que corre a experiência. A rotação do guiador foi captada com um TrackIR, que dispõe do seu próprio SDK (Software Development Kit).



Figura 7 – Bicicleta instrumentada.

No decorrer da experiência são fornecidas sensações visuais estereoscópicas através de um HMD. Este funciona também como mecanismo de entrada de dados, fornecendo a posição e rotação da cabeça do utilizador. O áudio binaural, fornecido através de auscultadores com cancelamento de ruído, permite a localização dos sons no espaço. A bicicleta está munida de um oscilador de contacto de baixa frequência que alimenta o veículo com as vibrações e impactos gerados pela física da simulação, nomeadamente ao nível da vibração causada pelo relevo do terreno. O cheiro típico do ambiente natural é também passado ao utilizador. A intensidade do vento resultante da deslocação e do ambiente natural da montanha é calculada em *software*, no entanto não temos disponível ainda o *hardware* de controlo respetivo, pelo que nesta fase apenas é utilizada uma ventoinha com intensidade fixa para simular o efeito.



Figura 8 – Vista da experiência na terceira pessoa.

Esta experiência contém apenas uma cena com as condições definidas pelo ambiente representado na tabela 1.

O terreno, árvores, cercas e vários outros conteúdos são objetos que contém componentes como sensores de colisão, que interagem com o motor de física para não permitir que sejam trespassados pela bicicleta. As flores de lavanda, tal como as árvores estão também munidas de um componente de cheiro que se fará sentir enquanto o utilizador estiver posicionado a uma distância delas em

que a atenuação do cheiro seja insuficiente para o cancelar.

Tabela 1 - Condições definidas pelo ambiente.

Propriedade	Valor
Place	Mountain
Time	Sunset
Weather	Cloudy, Dry
Gravity	Normal

A bicicleta é também um objecto (Figura 8). Este objeto tem como descendente o objeto ciclista, que além de ser visualizado é o avatar associado ao utilizador. A este utilizador estão por sua vez ligados renderizadores de imagem, áudio, cheiro e vento, que se encarregam de enviar as informações de cada sensação para o exterior. Os renderizadores trabalham em conjunção com os componentes aplicados aos objetos, sendo calculados em tempo real, por exemplo, a atenuação de cada cheiro ou som com base nas posições e outros parâmetros relativos à combinação entre renderizadores e componentes dos objetos. Na seguinte tabela apresentamos a relação entre diversos componentes e os renderizadores aplicados ao utilizador.

Tabela 2 - Componentes e respetivos renderizadores.

Renderizador	Componentes
Visual (Estereoscópico)	Meshes, Materiais (texturas/shader)
Auditivo (Binaural)	Fontes de som mono
Háptico	Sensores de colisão, Materiais (texturas de relevo)
Olfativo	Fontes de cheiro

Os materiais são conjugados com o renderizador háptico, para que a vibração e o impacto transmitidos ao dispositivo háptico dependem do relevo do solo no trajeto da bicicleta, codificado no material como um *heightmap*. Por outro lado, os sensores de colisão quando despoletados por outros objetos, enviam um sinal para o renderizador, que transmitirá impacto à bicicleta. Simultaneamente, sensores de colisão aplicados ao ciclista virtual fazem com que o renderizador háptico ative atuadores presentes num colete aplicado ao utilizador real sempre que se dá um choque entre o utilizador virtual e um objeto externo (um ramo de uma árvore, por exemplo).

A implementação da experiência foi realizada com base no modelo aqui apresentado, havendo por isso uma separação entre o *hardware* propriamente dito e a descrição dos estímulos (através dos renderizadores. Como exemplo, o projeto MASSIVE dispõe de três dispositivos diferentes para a emissão de cheiro: Scentroid, SensoryCo e um dispositivo genérico. O renderizador encarrega-se de redirecionar a saída para a máquina relevante.

O vento ambiental é renderizado com base nos parâmetros do ambiente, definidos na cena e não no utilizador. Por outro lado, a deslocação de ar sentida com a deslocação da bicicleta será calculada em tempo real. Para situações deste tipo, mais específicas, é utilizado o componente de scripting, que permite estender as propriedades e comportamentos dos objetos. O componente de scripting é também utilizado neste caso para filtrar o controlo da bicicleta. Por exemplo, ao virar o guiador, tanto o eixo frontal da bicicleta (também organizado numa estrutura hierárquica como filho do objeto bicicleta) como as mãos e braços do ciclista são alterados. Este é o tipo de casos em que o componente de *scripting* é útil: situações que necessitam de comportamentos bastante específicos. Será também uma forma de controlar objetos com base em entradas e complementa a simplicidade do modelo apresentado com a versatilidade quando necessária.

6. CONCLUSÕES

Neste artigo apresentámos um modelo de descrição de experiências multissensoriais com suporte para vários utilizadores. Da análise do trabalho relacionado, constatou-se o interesse e necessidade deste tipo de descrições, mas verificou-se que grande parte dos trabalhos se foca na componente dos dispositivos de interação (entrada) ou na componente do controlo dos dispositivos de estímulos (saída), e sem integrar estes componentes com suporte para vários utilizadores. Numa outra vertente, as arquiteturas baseadas em componentes utilizadas em vários dos atuais motores de jogo são bastante versáteis, mas não contemplam por princípio componentes multissensoriais que permitam a fácil criação desse tipo de experiências.

O modelo apresentado cobre estas componentes, mantendo uma camada de abstração que permite separar a descrição dos efeitos sensoriais a simular dos estímulos efetivamente transmitidos ao utilizador. Estes últimos são calculados em tempo real em função das especificações constantes da descrição da experiência, para a qual propomos também um formato de ficheiro baseado em XML. Finalmente, fornecemos um exemplo de aplicação do modelo proposto, em que se exemplificam os mapeamentos possíveis entre dispositivos, elementos de cena e utilizadores.

Espera-se que o modelo sirva como suporte para a criação e entrega de conteúdos multissensoriais, nomeadamente através da criação de ferramentas de autoria, que estão atualmente a ser desenvolvidas no contexto do projeto MASSIVE, no qual este trabalho se insere.

Em termos de trabalho futuro, pretende-se testar a *framework* através de um interpretador que está a ser desenvolvido para o efeito que permitirá compilar os dados de cenários criados por utilizadores. Pretende-se testar a ferramenta com utilizadores em contexto real como por exemplo profissionais em áreas como a psicologia ou a arquitetura e aferir resultados decorrentes das tarefas. Julgamos que será possível, após a otimização das ferramentas propostas, um elevado grau de autonomia por parte dos utilizadores no processo de autoria de cenários multissensoriais virtuais.

7. AGRADECIMENTOS

Este trabalho foi financiado pelo governo português através da FCT - Fundação para a Ciência e a Tecnologia e pela União Europeia (COMPETE, QREN E FEDER) no âmbito do projecto REC I/EEI-SII/0360/2012, com a designação “MASSIVE - Multimodal Acknowledgeable multiSenSorial Immersive Virtual Enviroments”.

8. REFERÊNCIAS

- [1] Oviatt, S., Cohen, P., Wu, L., Duncan, L., Suhm, B., Bers, J.,...Ferro, D. (2000). Designing the User Interface for Multimodal Speech and Pen-Based Gesture Applications: State-of-the-Art Systems and Future Research Directions. *Human-Computer Interaction*. doi:10.1207/S15327051HC11504_1
- [2] Irawati, S., Calderón, D., & Ko, H. (2005). Semantic 3D object manipulation using object ontology in multimodal interaction framework. *Proceedings of the 2005 International Conference on Augmented Tele-Existence*, 35–39.
- [3] Ob, S., & Hahn, M. (2010). The ontology for experience media authoring system. *Proceedings - 2010 3rd IEEE International Conference on Computer Science and Information Technology, ICCSIT 2010, 1, 207–210*. doi:10.1109/ICCSIT.2010.5564083
- [4] Gutierrez, M., Thalmann, D., & Vexo, F. (2005). Semantic virtual environments with adaptive multimodal interfaces. *Proceedings of the 11th International Multimedia Modelling Conference, MMM 2005*, 277–283. doi:10.1109/MMMC.2005.65
- [5] B. Xiao, R. Lunsford, R. Coulston, M. Wesson, and S. Oviatt. Modeling multimodal integration patterns and performance in seniors: toward adaptive processing of individual differences. In *Proceedings of the 5th international conference on Multimodal interfaces*, pages 265–272. ACM Press, 2003.
- [6] M. Waltl, C. Timmerer, H. Hellwagner, “A Test-Bed for Quality of Multimedia Experience Evaluation of Sensory Effects”, *Proceedings of the First International Workshop on Quality of Multimedia Experience (QoMEX 2009)*, San Diego, USA, July, 2009.
- [7] Timmerer, C., Waltl, M., Rainer, B., & Hellwagner, H. (2012). Assessing the quality of sensory experience for multimedia presentations. *Signal Processing: Image Communication*, 27(8), 909–916. doi:10.1016/j.image.2012.01.016
- [8] Waltl, M., Rainer, B., Timmerer, C., & Hellwagner, H. (2013). An end-to-end tool chain for Sensory Experience based on MPEG-V. *Signal Processing: Image Communication*, 28(2), 136–150. doi:10.1016/j.image.2012.10.009
- [9] C. Timmerer. A Summary of the Representation of Sensory Effects(RoSE). 2008.