

Realidade Misturada: Conceitos, Ferramentas e Aplicações

Ildeberto Aparecido Rodello¹
Silvio Ricardo Rodrigues Sanches²
Antonio Carlos Sementille, José Remo Ferreira Brega³

Resumo: A Realidade Misturada propõe a combinação de cenas do mundo real com o virtual, oferecendo ao usuário uma maneira intuitiva de interagir com determinada aplicação. Esse artigo tutorial tem como objetivo apresentar os conceitos fundamentais dessa forma emergente de interface entre homem e computador.

Palavras-chave: Realidade Misturada. Realidade Virtual. Interação Humano-Computador.

Abstract: *The Mixed Reality proposes scenes combining between virtual and real worlds offering to the user an intuitive way of interaction according to a specific application. This tutorial paper aims at presenting the fundamentals concepts of this emergent kind of human-computer interface.*

Keywords: *Mixed Reality. Virtual Reality. Human-Computer Interaction.*

1 Introdução

A interface entre homem e máquina se constitui em uma área de grande importância para a ciência da computação, uma vez que por meio dela acontece o relacionamento entre usuário e computador, o que pode decretar o sucesso ou o fracasso de determinado software ou hardware.

Nesse sentido, várias pesquisas buscam aprimorar e/ou criar dispositivos de entrada e saída [34], layouts, disposição e combinação de cores e formas de interação [35], entre outros. O objetivo principal é buscar uma maneira de o usuário conseguir utilizar o computador por meio de algum dispositivo, de uma forma que não sinta dificuldades e consiga extrair o máximo possível da capacidade da máquina.

Considerando uma simples escala evolutiva, a interface dos primeiros computadores valvulados se constituía de chaves e reles. A seguir é possível citar a interface por linha de comando, exemplificada especialmente por aquela oferecida pelo *prompt* do sistema operacional DOS (*Disk Operating System*). A interface do tipo WIMP (*Windows, Icons, Pointers and Menus*) ganhou impulso e notoriedade com o advento do sistema operacional Windows. Esse tipo de interface talvez predomine na maioria das aplicações até hoje desenvolvidas.

Contudo, novas formas de interface são estudadas com o objetivo de tornar a relação entre computador e usuário mais fácil, intuitiva e acessível a todos, incluindo nesse contexto os indivíduos portadores de necessidades especiais (PNE). Assim, estão em desenvolvimento interfaces baseadas em comandos de voz [36], interfaces tangíveis [35] e também interfaces de Realidade Misturada (RM), entre outras.

Realidade Misturada é um termo utilizado para caracterizar o espaço de transição entre o mundo real e um mundo artificial gerado por computador, denominado por muitos autores como Realidade Virtual (RV) [1]. Um ambiente de RM baseia-se na coexistência de objetos reais e virtuais, em que o usuário interage sem distinção,

¹ FEARP – USP – Avenida Bandeirantes, 3900 – Ribeirão Preto (SP) – Brasil.
{rodello@fearp.usp.br}

² Interlab – POLI – USP – Av Professor Luciano Gualberto, Travessa 3, número 158 – São Paulo (SP) – Brasil.
{silviorrs@gmail.com}

³ Departamento de Computação – Unesp – Caixa Postal 437 – Bauru (SP) – Brasil.
{remo, semente@dc.unesp.br}

de maneira fácil e intuitiva. Para muitos, a RM é considerada como a próxima geração de concepção de interfaces.

Nesse contexto, é objetivo deste artigo apresentar os principais conceitos relacionados à RM, bem como caracterizar todo o espaço de transição entre o real e o virtual. São apresentadas também algumas linguagens de programação e ferramentas que podem ser utilizadas para o desenvolvimento de aplicações de RM, além de algumas áreas que já apresentam potencial para a inserção dessa tecnologia.

Assim, este artigo está organizado da seguinte forma: a seção 2 aborda os conceitos básicos da RM, sendo apresentados caracterizações, dispositivos utilizados e diferenciações; as principais linguagens de programação e ferramentas para o desenvolvimento de sistemas de RM são apresentadas na seção 3; na seção 4 são discutidas, mas não esgotadas, algumas das possíveis áreas de aplicação; por fim, a seção 5 apresenta as considerações finais, que são seguidas das referências bibliográficas utilizadas na redação do texto.

2 Conceitos Básicos da Realidade Misturada

Historicamente, atribui-se o termo “realidade virtual” a Jaron Lanier, cientista da computação, artista visual e compositor, que no final da década de 80 sugeriu a junção de dois conceitos antagônicos para a criação dessa emergente área de pesquisas, que busca unir o real com o virtual.

Desde então, vários pesquisadores tentam caracterizar diferenciações entre a proporção de real e virtual presentes num determinado ambiente, além de cunhar termos para tanto. Uma dessas caracterizações refere-se ao *continuum* de virtualidade (Figura 1) proposto por [1].

2.1 Continuum de Virtualidade

Como mencionado, o *continuum* de virtualidade foi proposto por [1] para a caracterização dos ambientes de Realidade Misturada (RM). Pela Figura 1 é possível observar que nas extremidades são caracterizados os ambientes real e virtual. Quando há predominância do virtual sobre o real, denomina-se Virtualidade Aumentada (VA) e, com o contrário, Realidade Aumentada (RA). O conjunto, tanto VA quanto RA, caracteriza-se por RM.

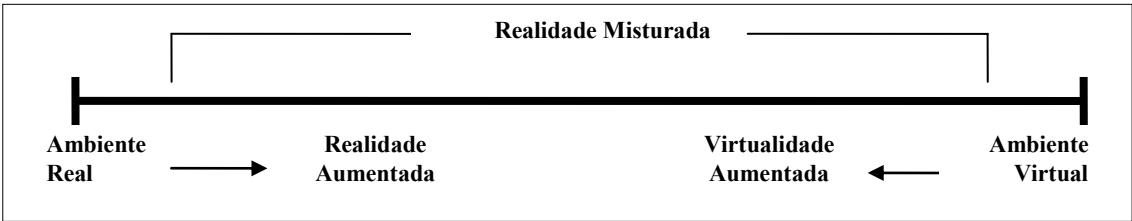


Figura 1 – Continuum de Virtualidade. Adaptado de [1].

Por ambiente virtual entende-se o ambiente totalmente gerado por computador, podendo ou não representar um ambiente já existente. Por ambiente real considera-se aquele que existe na realidade.

Realidade Melhorada é outra terminologia encontrada para caracterizar sistemas com características semelhantes. O conceito fundamental para este termo é a ideia de anotação visual [2]. Assim, informações dos objetos em cena, tais como tamanho e distância, entre outras características, podem ser descritas na tela como forma de melhoria do entendimento da mesma. Além de melhoria de conteúdo, o uso do processamento da imagem para gerar um aumento na sua qualidade também pode ser considerado Realidade Melhorada.

As definições mais antigas vinculam sistemas de RM à utilização de *Head-Mounted Displays* (HMDs) (Figura 2(a)). No entanto, esta é uma definição bastante restritiva e boa parte dos sistemas existentes não estariam enquadrados nesta definição. Nesse contexto, [3] apresentou uma ideia mais abrangente para a definição de RM. A proposta baseia-se na coexistência de três características essenciais para que um sistema seja considerado um sistema de RM. São elas:

- combinação de algo real e virtual;

- interação em tempo real;
- alinhamento e sincronização precisos dos objetos virtuais tridimensionais com o ambiente real (Registro).

2.2. Caracterização da Realidade Virtual

Baseada na coexistência dos conceitos de imersão, interação e envolvimento [4], a RV consiste em um ambiente tridimensional totalmente gerado por computador, denominado Ambiente Virtual (AV), dentro do qual o usuário pode agir de forma intuitiva e idêntica ao seu cotidiano, apto a atuar de acordo com seis graus de liberdade: rotação e translação nos eixos cartesianos X, Y e Z.

Esses conceitos são alcançados, principalmente, com a utilização de dispositivos visuais, auditivos e físicos especiais (denominados não convencionais). Entretanto, nada impede que também possam ser obtidos com os dispositivos convencionais, não tendo, nesse caso, toda sua potencialidade utilizada.

A ideia de imersão é ligada ao sentimento de estar dentro do AV gerado. Normalmente, um sistema classificado como imersivo é obtido com o uso de dispositivos visuais especiais, como capacetes de visualização (HMD) (Figura 2(a)) e óculos estereoscópicos (*Stereoglasses*, *Shutterglasses*) (Figura 2(b)) associados a dispositivos de rastreamento e captura de movimentos (*Body Tracking*). Tais dispositivos possibilitam a visão tridimensional do ambiente gerado, de maneira similar a um ambiente real.



Figura 2 – Exemplos de *Head-Mounted Displays* [5] e *Stereoglasses*

O princípio de utilização dos capacetes consiste em projetar as imagens do ambiente virtual diretamente nos olhos do usuário por meio de dois pequenos monitores. Dessa forma, os visores do capacete tornam-se os “olhos” do indivíduo dentro do mundo. Além disso, todos os movimentos da cabeça também podem ser transferidos ao ambiente, como se a pessoa estivesse realmente dentro daquele ambiente.

A imersão por meio dos óculos estereoscópicos, por sua vez, é alcançada em conjunto com monitores, mesas especiais ou salas com projeção das visões nas paredes, piso e teto. A utilização de óculos juntamente com monitores é o mais comum, devido principalmente ao baixo custo associado. Nesse conjunto, as imagens são vistas de maneira estereoscópica na tela.



Figura 3 – Exemplos de *Responsive Workbench* [6] e *Caverna Digital* [7]

As mesas especiais, também denominadas *Responsive Workbenches* [8], (Figura 3(a)), consistem em uma plataforma com projeções na sua superfície e, com o auxílio dos óculos estereoscópicos, as imagens são vistas com efeitos de profundidade. Por fim, os sistemas imersivos baseados em salas com projeção das visões nas paredes, piso e teto, denominado Cavernas Digitais (CAVE - *Cave Automatic Virtual Environment*) [9] [10] (Figura 3(b)), permitem ao usuário um alto grau de imersão, em que todos seus movimentos são capturados e transferidos para o ambiente. Este modelo, entretanto, tem alto custo.

Na ausência dos dispositivos especiais, a RV é classificada como não imersiva, visto que o usuário somente visualiza o ambiente virtual de forma tridimensional diretamente na tela do monitor, interagindo com o ambiente por meio de dispositivos convencionais, como mouse ou teclado. Estes, porém, limitam as ações dentro do ambiente. A RV não imersiva pode ser chamada de Sistemas de Janelas no Mundo (WoW - *Window on World System*), ou ainda RV de Mesa [11].

A ideia de interação é ligada com a capacidade do computador de detectar as entradas do usuário e de modificar instantaneamente o mundo virtual e as ações sobre ele. A detecção dessas entradas também pode estar associada ao uso de dispositivos especiais, como luvas, *spaceballs*, *joysticks* e mouses3D, além dos convencionais mouse e teclado.

Munido de luvas (Figura 4), o usuário ganha a capacidade de tocar, apontar, manipular e até sentir algum objeto no mundo virtual. As luvas fazem o papel de uma mão virtual dentro do ambiente, permitindo ao usuário realizar ações que serão respondidas pelo sistema. Além disso, os dispositivos físicos especiais podem trazer algum *feedback* para o usuário. Os *feedbacks* podem ser classificados em tátil, de força ou térmico.



Figura 4 – Exemplo de Luvas [5]

No *feedback* tátil são transmitidas para o usuário sensações táteis como a textura de uma determinada superfície. No *feedback* de força, o dispositivo utilizado se contrapõe a uma determinada força aplicada pelo usuário, oferecendo resistência e representando variáveis físicas como peso. Por fim, o *feedback* térmico busca representar as sensações de calor e frio.

A ideia de envolvimento, por sua vez, é ligada com o grau de motivação para o engajamento de uma pessoa com determinada atividade. O envolvimento pode ser passivo, como passear por um museu, ou ativo, como participar do desenvolvimento de um determinado projeto com algum parceiro. A RV tem potencial para os dois tipos de envolvimento ao permitir a exploração de um ambiente virtual e ao propiciar a interação do usuário com o mundo virtual dinâmico.

2.2.1. Classificações

São várias as formas pelas quais um AV pode ser classificado. Vários autores propõem formas de classificação, não existindo uma aceita como sendo a única ou a principal. As classificações aqui apresentadas são as encontradas com mais frequência na literatura.

Os AVs podem ser classificados de acordo com o modelo de participação do usuário, sendo classificadas em passiva, exploratória e interativa [11].

Em uma participação passiva, o AV permite ao usuário somente a exploração predefinida, sem a possibilidade de interferência. A rota e o ponto de vista são controlados pelo software. Em um AV que permita a participação exploratória, o usuário tem a liberdade de escolher a rota e o ponto de vista, entretanto não tem liberdade para interação com o ambiente. A participação interativa, por sua vez, oferece todas as possibilidades que eram restritas nas participações passiva e exploratória. O usuário tem liberdade de escolher rota, ponto de vista, e de interagir com os objetos do ambiente.

É possível ainda classificar os AVs de acordo com o grau de imersão proporcionado ao usuário. De acordo com essa classificação, têm-se RV de simulação, RV de projeção, Displays Visualmente Casados e RV de mesa [11].

A RV de simulação é aquela tida como a mais antiga. Ambientes desse tipo reproduzem um simulador (carro, avião, etc.), oferecendo ao usuário um ambiente similar à simulação em questão. Na RV de projeção, o usuário não tem uma representação dentro do AV, mas pode se comunicar com objetos ou representações nele

contidos. É também conhecida com Realidade Artificial. Quando o usuário se equipa com dispositivos que lhe apresentam imagens exibidas diretamente nos olhos e que contém sensores que acompanham os movimentos de sua cabeça, tem-se o tipo Displays Visualmente Casados. Na RV de mesa, por sua vez, são utilizados grandes monitores ou algum sistema de projeção para exibição do AV. É um tipo de RV que classifica os ambientes que não possuem dispositivos especiais.

2.3. Caracterização da Realidade Aumentada

De acordo com [3], a Realidade Aumentada (RA) é uma variação da RV. Supõe-se que a RV proporciona a completa imersão do usuário em um ambiente sintético totalmente gerado pelo computador (AV) e, uma vez imerso, ele não consiga ver o ambiente real. A RA permite que o usuário visualize os objetos virtuais sobrepostos ou compondo uma cena com o mundo real. Dessa forma, objetos reais e virtuais coexistirão num mesmo espaço (Figura 5).

Como destacado por [12], a RA é uma melhoria do ambiente real com textos, imagens e objetos virtuais gerados por computador. Em termos de interface, a RA diferencia-se da RV por aumentar a percepção do usuário e, consequentemente, contribuir para que a interação seja mais intuitiva. Visa melhorar o retorno natural que o usuário tem. A utopia é a criação de um ambiente em que o usuário não consiga distinguir o mundo real do virtualmente aumentado.



Figura 5 – Exemplos de Ambientes de RA [13] [14]

Ainda comparando RV e RA, [15] destacam que: i) a RV é gerada por computador. A RA, por sua vez, visa enriquecer o ambiente real com objetos virtuais; ii) na RA o usuário tem o sentido de presença no ambiente real, ao passo que na RV a sensação visual é controlada pelo computador; iii) para que a RA funcione é necessário um dispositivo para combinar o real com o virtual. Na RV é preciso um dispositivo para proporcionar a imersão do usuário ao ambiente virtual.

Quanto a dispositivos específicos, com destaque para a visualização, [12] apresenta duas categorias: *video displays* e *optical displays*. A categoria *video displays* restringe a área de visualização ao espaço do monitor por meio de imagens geradas por uma câmera. Na categoria *optical displays*, que [1] se caracterizam como *head-mounted see-through*, permite-se ao usuário “ver através” do display. Geralmente se constitui de visores transparentes, que possibilitam a sobreposição de imagens virtuais à visão do ambiente real.

A simplicidade é uma das vantagens que é possível se destacar dos *video displays*. Os objetos virtuais podem ser facilmente inseridos por meio da alteração da *stream* de vídeo que será exibida. A resolução da câmera e dos dispositivos de visualização é ponto fraco dessa categoria. A visualização direta do ambiente real tende a ser melhor naturalmente. Outro ponto fraco refere-se à fadiga causada pela visualização no monitor durante longos períodos [12].

O ponto fraco dos *optical displays* é a dificuldade de registro do ambiente, ou seja, o perfeito alinhamento do objeto virtual no ambiente real. Nesse tipo de *display* a liberdade do usuário é maior e, por consequência, o ambiente é menos controlado, o que causa dificuldades na composição da cena. Além disso, o custo desse tipo de dispositivo é um grande obstáculo.

Novos conceitos de dispositivos de visualização que são considerados uma variação da tecnologia tradicional, apresentados por [15], foram denominados *Spatial Augmented Reality* (SAR). Esses dispositivos são caracterizados principalmente pelo fato de estarem separados do corpo do usuário e não acoplados à cabeça

como os tradicionais. Existem três diferentes abordagens relacionadas a estes dispositivos, as quais são diferenciadas principalmente pelo modo como os objetos virtuais são inseridos no ambiente.

Os dispositivos de exibição por meio de vídeo baseados em telas (*screen-based video see-through displays*) são muitas vezes considerados uma janela do mundo virtual. Nesta categoria se encontram os monitores convencionais, nos quais são exibidas as imagens misturadas. Embora forneçam um baixo grau de imersão, devido ao campo de visão estar limitado ao tamanho do monitor, aplicações que utilizam este tipo de dispositivo são bastante comuns e estão dentro do contexto de RM.

Os *spatial optical see-through displays*, contrariamente aos dispositivos presos à cabeça ou portáteis, geram imagens que são ajustadas dentro do ambiente real. Este tipo de combinação óptica é representado pelas telas transparentes, hologramas ópticos, entre outras tecnologias de exibição de imagens similares, que possuem recursos de visualização não acoplados ao observador.

Nos *projector-based spatial displays* as imagens não são formadas em seus próprios *displays*, mas por meio de uma projeção direcionada para construí-las diretamente sobre a superfície de objetos físicos no campo de visão do observador. Existem aplicações que suportam projetores estáticos ou dirigidos e, em alguns casos, múltiplos projetores são utilizados para aumentar a área passível de projeção.

2.3. Caracterização da Virtualidade Aumentada

De acordo com [16], a VA pode ser caracterizada como uma particularização da RM, quando o ambiente virtual predomina em relação ao real. É caracterizada também pelo enriquecimento do ambiente virtual com objetos reais que podem ser capturados em tempo real.

2.4 Componentes de um Sistema de Realidade Misturada

A obtenção de um modelo de sistema para RM torna-se uma tarefa bastante complicada devido a infinidade de dispositivos e aplicações existentes. Apesar disso, é possível, analisando-se suas características, a identificação de tarefas essenciais que podem ser organizadas e classificadas como sistemas menores (subsistemas).

A proposta apresentada por [17] em complemento àquela apresentada por [18] sugere, pelo menos, cinco subsistemas considerados indispensáveis a qualquer sistema de RA e VA. Nesses subsistemas são divididas as tarefas de captação da imagem real, obtenção da posição e orientação do subsistema de captura da imagem real, geração de objetos virtuais, mistura da imagem e exibição da imagem de saída. Estendendo um pouco mais este conceito para considerar a existência de objetos reais em movimento na cena, fato comum em muitas aplicações de RM, [18] propuseram o acréscimo de três outros subsistemas: um subsistema de manipulação real de objetos que pode lidar tanto com os objetos reais quanto com os virtuais, um subsistema para obtenção da localização do subsistema citado e um subsistema para obtenção da localização dos objetos reais que se movimentam no ambiente aumentado.

O primeiro subsistema adicional, segundo [18], deve ser incorporado aos apresentados por [17], e os dois últimos, mesclados e transformados em um único para localização global. Na Figura 6 é mostrada a arquitetura de um sistema de RM típico de acordo com o modelo proposto por [18].

Um sistema de Realidade Misturada típico pode, então, ser caracterizado pelos seis subsistemas mostrados na Figura 6. De maneira simplificada, é possível definir o Subsistema de Captura do Mundo Real como o sistema que estimula os respectivos sentidos do ser humano com sinais originados do mundo real.

O Subsistema Gerador de Modelos Virtuais é responsável pela construção da parte gráfica. Por meio deste subsistema é possível classificar os modelos gráficos gerados analisando-se o grau de realismo, o modo como são construídos, níveis de interação com o usuário e o tipo de tecnologia utilizada.

O Subsistema Misturador de Realidades realiza a combinação do real e do virtual e permite a identificação das tecnologias de mistura que podem ser ópticas ou eletrônicas.

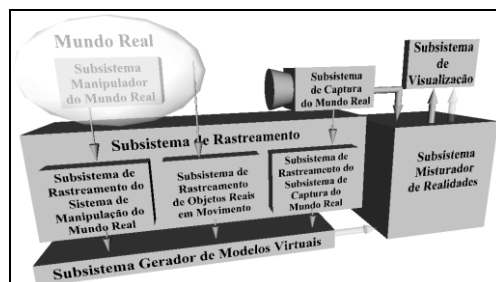


Figura 6: Componentes de Um Sistema de Realidade Aumentada [18]

O Subsistema Manipulador do Mundo Real possui características completamente diferentes para cada tipo de tecnologia utilizada com esta finalidade. É responsável pela interação háptica com o mundo real ou virtual.

O Subsistema de Rastreamento é o responsável pela obtenção da localização do observador e dos objetos reais de forma precisa. Este subsistema pode ser caracterizado por características como: grau de complexidade e tipo de rastreamento (passivo, ativo ou híbrido). O rastreamento passivo pode ser realizado por meio de métodos que utilizam marcadores [37] [38] dispostos no ambiente para auxiliar a estimativa da posição e orientação dos elementos da cena [39], ou por métodos capazes de extrair tais coordenadas a partir de características do próprio ambiente (sem marcadores) [40]. O rastreamento ativo utiliza equipamentos, como GPS, que emitem algum tipo de sinal [41]. Subsistemas híbridos também são utilizados para tornar o processo mais robusto [42].

O Subsistema de Visualização está relacionado com a tecnologia dos dispositivos utilizados para esta finalidade. É responsável pela identificação do ponto de vista do usuário, que pode estar totalmente imerso pelo uso do HMD ou enxergando o mundo em uma janela de monitor de vídeo convencional. A quantidade de cores suportadas pelos dispositivos e as taxas de quadros exibidas podem ser obtidas por meio deste subsistema.

3 Linguagens, Bibliotecas e Ferramentas para Realidade Misturada

Existem várias possibilidades em termos de linguagens de programação e plataformas de desenvolvimento para a implementação de ambientes de RM. Esta seção tem como objetivo apresentar superficialmente algumas dessas alternativas.

3.1. Open GL

OpenGL (*Open Graphics Library*) é uma biblioteca gráfica de baixo nível, que oferece ao desenvolvedor um pequeno conjunto de primitivas geométricas (*points*, *line segments*, ou *polygons*), além de um conjunto de comandos que possibilitam a especificação de objetos em duas ou três dimensões, usando as primitivas. A base da OpenGL são as linguagens de programação C/C++; assim, seu funcionamento é semelhante ao de uma biblioteca C.

Segundo [19], a linguagem OpenGL funciona como uma máquina de estados, onde o controle de vários atributos é realizado por meio de um conjunto de variáveis de estado que, inicialmente, possuem valores padrão, podendo ser alterados caso seja necessário. Por exemplo, todo objeto será traçado com a mesma cor até que seja definido um novo valor para esta variável.

De maneira geral, OpenGL se compõe de duas bibliotecas: *Opengl32.lib* ou *libGL.so* e *OpenGL Utility Library*. As primeiras referem-se a arquivos com configurações do *driver* da placa de vídeo, enquanto a última (que pode ser *glu32.lib*, *libGLU.so* ou *libMesaGLU.so*) oferece funções para o desenvolvimento de aplicações.

As aplicações OpenGL executam de acordo com o *pipeline* de funcionamento apresentado na Figura 7. Inicialmente, todas as funções obedecem a uma configuração inicial, que permanece intacta até que haja a alteração de algum dos seus parâmetros.

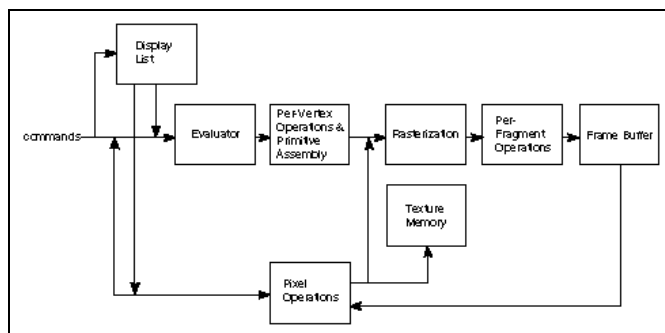


Figura 7 – Pipeline básico de funcionamento da OpenGL [20].

No estágio *evaluator* são fornecidos meios para aproximação de curvas e superfícies geométricas. No próximo estágio (*per-vertex operations and primitive assembly*) ocorre o processamento das primitivas geométricas. No estágio *rasterization* é produzida uma série de endereços de *frame buffer* e são associados valores usando uma descrição bidimensional de um ponto, de um segmento de linha ou de um polígono. Cada fragmento produzido é enviado para o último estágio (*per-fragment operations*), que executa as operações finais antes de colocá-lo como um *pixel* no *frame buffer*.

3.2. Virtual Reality Modeling Language

Com sua primeira versão lançada em 1994, *Virtual Reality Modeling Language* (VRML) é uma linguagem de propósito geral para a descrição de cenas 3D (tridimensionais), usando uma grande quantidade de nós gráficos preestabelecidos. Tais nós são organizados em uma estrutura hierárquica denominada grafo de cena (*scene graph*). A última versão da VRML foi padronizada pela ISO (ISO/IEC 14772:1997) Alguns desses nós são:

Behaviors: Refere-se às mudanças feitas na estrutura ou na aparência de uma cena 3D. VRML fornece mecanismos para animação local ou remota de qualquer ramo componente do grafo de cena.

Shape: Este nó é um recipiente que coleciona um par de componentes, chamados de geometria e aparência. Eles descrevem objetos que tipicamente são usados nos campos de geometria e aparência de uma forma 3D.

Geometria: *Box*, *Cone*, *Cylinder* e *Sphere* são poliedros regulares simples (também chamados de primitivas) que fornecem as formas básicas predefinidas para a construção de um objeto. O nó *Text* simplifica a especificação de um texto planar ou poligonal. Além disso, é possível destacar outras primitivas que facilitam a construção de geometrias: *Elevation Grid*, *Extrusion*, *IndexedFaceSet*, *IndexedLineSet* e *PointSet*, que criam geometria 3D com certa complexidade.

Aparência: A aparência de uma geometria é controlada pela especificação das cores ou textura. Nesse contexto, é possível destacar algumas primitivas, tais como *Material*, que permite a especificação de cores, além do grau de transparência. Com primitiva *ImageTexture* é possível aplicar uma textura 2D sobre uma imagem ou objeto

Topologia da Cena: Agrupamento e nós *Child*: A sintaxe VRML força uma estrutura hierárquica (grafo de cena), de forma que os navegadores (*browsers*) podem executar uma renderização eficiente e otimizações computacionais. O agrupamento é usado para descrever as relações entre formas (*Shapes*) e outros nós *Child*. Os nós *Child* descrevem iluminação, sons, visão, sensores de ação e interpoladores de animação.

Agrupamento: É usado para descrever as relações espaciais e lógicas entre os nós. Dessa forma, habilita-se uma renderização eficiente em navegadores 3D. A primitiva mais simples para o agrupamento é a *Group*. O nó *Transform* simplesmente aplica rotação, escala e translação. O nó *Collision* permite que o autor especifique uma caixa em torno do objeto para ajudar nos cálculos de detecção de colisão.

Agrupamento e Web: Dois tipos de agrupamento são possíveis em cenas 3D na Web: o *Inline* permite a importação de dados 3D de um outro mundo VRML; o *Anchor*, por sua vez, cria um *link* entre o nó *child* e uma URL. Assim, quando se clica em um nó especificado como *Anchor*, a cena VRML atual é substituída por uma contida na URL indicada.

Iluminação e Sons: Para determinar a iluminação, luzes virtuais são adicionadas à cena 3D. As mesmas são usadas para calcular a visibilidade, brilho e reflexão de acordo com um modelo matemático de iluminação especificado. As primitivas usadas para tanto são *DirectionalLight*, *PointLight* e *SpotLight*.

Visão: A maioria dos nós 3D descreve a localização, tamanho, forma e aparência de um modelo. O nó *Viewpoint* especifica a posição e orientação de um campo de visão para uma “câmera virtual”, que será usada para visualizar a cena 3D e renderizar a imagem na tela.

Sensores de Ação: Os sensores detectam a mudança na cena com a passagem do tempo (*TimeSensor*), intervenção do usuário ou outra atividade como proximidade de visualização (*VisibilitySensor*). Os valores obtidos por tais sensores podem ser redirecionados a eventos de entrada ou saída conectados a campos na cena via ROUTEs.

Interpoladores de Animação: O ponto principal de uma animação consiste, basicamente, de uma simples variação de valores com relação aos tempos aplicados a campos de um determinado nó. A interpolação linear é utilizada em cores, coordenadas, orientação, posicionamento e campos escalares.

Protótipos: Permitem a criação de um novo mundo VRML por meio da combinação de nós e campos de outro tipo de nó já existente. Nesse contexto, existem duas primitivas: PROTO, que é algo análogo a uma macro, e EXTERNPROTO, que especifica uma URL remota onde o PROTO original está definido.

3.3. X3D

Tida como o sucessor da VRML, a linguagem de marcação X3D também é padronizada (ISO/IEC 19775:2004). Sua principal característica é a capacidade de codificar a cena usando a sintaxe da XML (*eXtensible Markup Language*). Os principais objetivos adotados no projeto X3D foram:

- Separar a arquitetura de execução da codificação dos dados.
- Suportar grande variedade de formatos de codificação, incluindo XML.
- Adicionar novos objetos gráficos, comportamentais e interativos.
- Fornecer APIs alternativas para as cenas 3D.
- Definir subconjuntos de especificação (*profiles*).
- Eliminar, quando possível, comportamentos desnecessários.

X3D tem um grande conjunto de características para suportar uma variedade de aplicações. Tais características incluem:

- Gráficos 3D – Geometrias poligonais e paramétricas, transformações hierárquicas, iluminação, materiais e mapeamento de textura *multi-pass/multi-stage*.
- Gráficos 2D - Texto, vetores 2D e formas planares exibidas dentro de uma transformação 3D hierárquica.
- Animação - *Timers* e interpoladores orientam animações contínuas, animações humanóides e *morphing*
- *Spatialized* áudio e vídeo – Fontes audiovisuais são mapeadas para geometrias na cena.
- Interação - *Mouse-based picking* e *dragging*; entrada via teclado.
- Navegação – Câmeras, movimento do usuário na cena 3D, colisão, detecção de proximidade e visibilidade.
- Objetos definidos pelo usuário - habilidade de estender as funcionalidades nativas do browser pela criação de tipos de dados definidos pelo usuário.
- *Scripting* – Habilidade de mudar a cena dinamicamente via linguagens de programação e *scripting*
- Rede - Habilidade de compor uma cena X3D com objetos e recursos localizados na *World Wide Web*.

- Simulação Física – Animação humanoide; conjuntos de dados geoespaciais; integração com protocolos DIS (*Distributed Interactive Simulation*).

3.4. Java 3D

Java 3D é uma API inicialmente desenvolvida pela *Sun Microsystems* com o objetivo de ser uma interface para o desenvolvimento e apresentação de programas em Java com conteúdo tridimensional. Segue o paradigma de escreva uma vez e rode em qualquer lugar (*write once, run anywhere*), fornecendo suporte para várias plataformas, vários dispositivos de visualização e vários dispositivos de entrada.

Um programa Java 3D cria instâncias de objetos Java 3D que são colocados em uma estrutura hierárquica chamada de grafo de cena (*scene graph*). Esta estrutura segue um modelo que especifica o conteúdo de um universo virtual e como este conteúdo será renderizado.

Em um grafo de cena são definidas a geometria, sons, luzes, localização, orientação e aparência dos objetos, os quais são organizados de acordo com o modelo hierárquico de relacionamento. Como toda estrutura baseada em grafos, o grafo de cena é composto por nós (*nodes*) e arcos (*arcs*). Um nó irá representar um elemento e um arco, o relacionamento entre os elementos.

Como principais vantagens é possível citar [21]:

- Implementação somente em Java: é interessante desenvolver o código da aplicação, a persistência e a interface do usuário em somente uma linguagem, que ainda oferece facilidades para a portabilidade.
- Suporte para aplicações distribuídas: a linguagem Java é sinônimo de desenvolvimento para internet. *Applets* e *servlets* podem ser desenvolvidos com facilidade em Java. Além disso, um suporte de comunicação pode ser alcançado com Java RMI (*Remote Method Invocation*), fornecendo independência de plataforma e níveis de segurança.
- Descrição da cena: o modelo de grafo de cena facilita o entendimento, o reuso, o desenvolvimento e a manutenção da aplicação, onde objetos são representados e controlados por nós.
- Suporte para dispositivos não convencionais: Java 3D oferece suporte para dispositivos não convencionais, como capacetes de visualização (HMDs) e projetores para Cavernas Digitais.
- Abstração dos mecanismos de renderização: possui mecanismos para otimização de renderização dos grafos de cena.

Ainda de acordo com [21], Java 3D também oferece algumas desvantagens, dentre as quais é possível citar:

- algumas características oferecidas por linguagens gráficas mais apuradas como OpenGL podem não ser alcançadas em Java 3D;
- a abstração dos mecanismos de renderização também pode não ser desejável para desenvolvedores que necessitam de maior controle do grafo de cena;
- o coletor automático de lixo (*garbage collector*) nativo da linguagem Java pode depreciar o desempenho da cena;
- a distribuição da aplicação para os usuários pode se tornar difícil, uma vez que são necessários o *download* e a instalação da biblioteca Java 3D. Para usuários inexperientes, este pode ser um grande obstáculo.

3.5. ARToolkit

A biblioteca de software da linguagem C e C++ ARToolkit oferece um conjunto de funções que realizam algumas das tarefas mais complexas das aplicações de RA. Sua implementação é baseada em algoritmos de visão computacional para resolver o problema do cálculo do ponto de vista do usuário [22].

Este cálculo é uma das dificuldades clássicas no desenvolvimento de aplicações deste tipo e uma solução precisa é fundamental para que se possa renderizar a imagem virtual na posição e na orientação correta.

O projeto da biblioteca está sob a responsabilidade da Universidade de Osaka, no Japão, apoiada pelo Human Interface Technology Laboratory (HITLab) da Universidade de Washington, nos Estados Unidos, e pelo HITLabNZ na Universidade de Canterbury, na Nova Zelândia [23].

O ARToolkit é distribuído livremente para fins não comerciais sob a licença GNU, é suportado por várias plataformas e possui excelente documentação, sendo utilizado em conjunto com alguns softwares gráficos. A renderização normalmente é realizada pela biblioteca gráfica OpenGL ou OpenVRML (Open Virtual Reality Modeling Language) e a visualização pode ser feita por meio de dispositivos do tipo Optical ou Video See-Through [22].

Bibliotecas de alto nível de acesso [24] [25] também podem executar a renderização de objetos virtuais, e outras bibliotecas gráficas para auxiliar a construção de ambientes mais complexos podem ser utilizadas em conjunto [26].

Entre as principais funcionalidades do ARToolkit, é possível destacar o rastreamento da posição e da orientação da câmera, a utilização de simples quadrados marcadores para localização do posicionamento, a facilidade de configuração da câmera e uma execução rápida o suficiente para aplicações de tempo real.

As técnicas de visão computacional utilizadas pelo ARToolkit permitem o cálculo do ponto de visão da câmera de vídeo em relação a um marcador real colocado na cena. Esse processo inicia-se com a captura da imagem de vídeo em tempo real (Figura 8(a)). Essa imagem é convertida em imagem binária, obedecendo a um valor limite (*thresholding*) estabelecido para transformar as cores da imagem em preta ou branca (Figura 8 (b)) [22]. Este valor limite é fixado, mas pode ser alterado manualmente em tempo de execução, provocando uma interrupção no programa. A variação deste valor muitas vezes é necessária principalmente em ambientes em que a iluminação não é constante.

Na imagem binária são identificadas regiões quadradas, que passam por um processo de comparação com modelos de marcadores previamente inseridos no sistema. Havendo sucesso na comparação, um marcador foi identificado.

O fato de as dimensões do marcador estarem armazenadas no sistema juntamente com a orientação correta do identificador do usuário em seu interior, permite ao ARToolkit calcular a posição da câmera de vídeo em relação a este marcador físico. Estes valores são utilizados para encontrar as coordenadas para projeção dos objetos virtuais na tela, que deve ser feita precisamente sobre o marcador real (Figura 8 (c)) por meio de algum software gráfico [22].

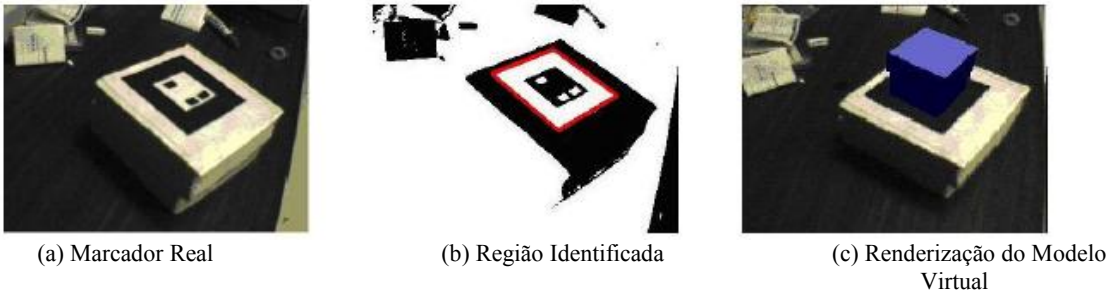


Figura 8: Processo de Reconhecimento [22]

3.6. JARToolkit

Uma alternativa para acessar as funções do ARToolkit utilizando as linguagens de programação Java, Lingo ou qualquer linguagem de script é a JARToolkit [27]. Baseada na JNI (*Java Native Interface*), possibilita, inclusive, trabalhar-se com bibliotecas de alto nível de acesso, como Java 3D, ou de baixo nível de acesso, como GL4JAVA, para renderização de objetos virtuais.

3.7. Outros

Além dessas linguagens, bibliotecas e ferramentas apresentadas, existem alternativas para a implementação de RM, incluindo nesse contexto *engines* usadas para o desenvolvimento de jogos. Recomenda-se a leitura de [28] para uma cobertura mais completa.

4 Aplicações

As áreas de aplicação da RM estão em constante expansão. Podem ser destacados projetos nas áreas de saúde (medicina), na manutenção de equipamentos, na engenharia biomédica e no treinamento militar, além de aplicações colaborativas de videoconferência ou voltadas ao entretenimento [3].

Isso acontece devido, principalmente, ao grande poder de visualização oferecido pela RM e também pela diminuição do custo de dispositivos não convencionais. Atualmente existem soluções tais como [29], que propõem a implementação de sistemas de projeção a custo reduzido.

4.1 Educação e Treinamento

Por sua forma de interação intuitiva e pelo seu grande poder de ilustração, a RM apresenta grande potencial para o desenvolvimento de aplicação com fins educacionais ou de treinamento.

Já existem propostas de ambientes de ensino/aprendizagem para áreas tradicionais, como química, física e matemática, além de soluções para treinamentos específicos. Há uma discussão interessante sobre o assunto em [30].

4.2 Aplicações Médicas

Nessa área, os conceitos de RM são aplicados para diversas finalidades. A mais explorada no momento é o treinamento médico. Devido à sua capacidade de simulação, a RM está sendo utilizada para treinar procedimentos que outrora eram testados em cadáveres ou cobaias.

Outra finalidade interessante refere-se ao planejamento de cirurgias ou cirurgias orientadas por conceitos de RM. Nesse cenário, o que é explorado é o seu poder de visualização, onde imagens virtuais podem ser sobrepostas a imagens reais. Uma abordagem mais aprofundada sobre o assunto pode ser encontrada em [31].

Em ortopedia e fisioterapia, a RM é usada na captura e análise de movimentos de pacientes [43]. Nesta situação, os movimentos de um paciente são capturados por uma câmera de vídeo e, por meio de técnicas de visão computacional, o movimento é transformado em dados computacionais para que o médico o analise. Esses dados podem ser armazenados para que futuramente o ortopedista possa acompanhar a evolução do tratamento.

4.3 Jogos e Entretenimento

Devido ao seu grande poder de motivação e envolvimento, os conceitos de RM cada vez mais têm sido utilizados não somente para a elaboração e implementação de jogos eletrônicos, mas também para a geração de recursos especiais em filmes. É possível considerar essa área como uma das mais promissoras, com grande potencial para investimentos.

Os usuários são atraídos por recursos gráficos tridimensionais cada vez mais apurados e por interações mais sofisticadas. [32] fazem uma vasta análise sobre o assunto.

4.4 Manufatura e Reparo

Manutenção e reparo são duas outras áreas em que a RM é aplicada. Um técnico, ao realizar a manutenção em um equipamento desconhecido, pode utilizar um display de RM para aumentar a imagem do equipamento com anotações e informações necessárias para se realizar as etapas da manutenção. Por exemplo, é

possível destacar as partes do hardware que deverão ser removidas para a realização do reparo e, então, a visão interna da máquina pode destacar as placas que deverão ser trocadas [17].

4.5 Anotação e Visualização

Outra aplicação da RM é na anotação e visualização de objetos e ambientes públicos ou privados. O usuário portaria um display do tipo handheld ou um optical see-through HMD para visualizar dicas, como, por exemplo, o ARLib [44], que permite localizar determinado livro em uma biblioteca com o auxílio de um notebook, um HMD com uma câmera de vídeo e o software Studierstube [45]. Esta aplicação tem o objetivo de enriquecer o ambiente real com informações sobrepostas às estantes da biblioteca. O livro desejado é destacado dos demais quando aparece no campo de visão do usuário.

4.6 Robótica

Na robótica, a RM pode ser aplicada no auxílio à movimentação de robôs a distância. Caso seja utilizada uma única câmera de vídeo para capturar a imagem do robô real, o sentimento de profundidade é perdido, podendo ocasionar acidentes ou erros. Porém, com o auxílio da RM o usuário pode planejar e executar os passos em um robô virtual sobreposto à imagem do robô real e, caso a representação virtual consiga executar a tarefa, o comando pode ser dado ao robô real, evitando ocorrência de erros, além de eliminar possíveis oscilações devido ao atraso da conexão [3].

4.7 Militar

Na área militar, a RM pode ser aplicada em treinamentos e em combate. Em um avião ou simulador de voo, é possível projetar dados sobre inimigos ou do próprio avião em um Head-up Display (HUD), que é uma tela transparente posicionada em frente ao piloto, oferecendo uma visão aumentada do mundo real. Um exemplo de HUD é o desenvolvido pela Elbit Systems [46].

5 Considerações Finais e Perspectivas

Este artigo teve como objetivo apresentar os principais conceitos relacionados ao desenvolvimento básico de aplicações de RM. Procurou-se apresentar e discutir caracterizações, classificações, dispositivos, ferramentas e possíveis áreas de aplicação, procurando não esgotar o tema, mas despertar o interesse e abrir discussões mais profundas sobre ele.

Como mencionado por [33], apesar da nítida evolução tanto tecnológica quanto orçamentária que a RM experimentou nos últimos tempos, existem desafios a serem superados, tais como:

- Melhoria no realismo das modelagens e simulações, principalmente para aplicações na área de saúde.
- Melhoria da capacidade reativa das aplicações, ou seja, do tempo de resposta em uma interação, por exemplo. Este é um fator primordial para a motivação do usuário dentro ambiente.
- Diminuição do custo para aquisição de dispositivo, apesar de já haver uma nítida diminuição, algumas soluções ainda tem alto custo agregado, o que torna sua utilização impraticável para os centros de pesquisas e universidades.
- Melhoria das interfaces classificadas como multimodais, isto é, interfaces que conjugam mais que uma forma de interação simultaneamente, tais como gestos e voz.
- Aprimoramentos na captura de imagens e rastreamento de movimentos do usuário, principalmente com o uso de dispositivos móveis.
- Refinamentos no tratamento de erros, principalmente nos erros de registro em ambientes de RM, ou seja, a renderização mais precisa do objeto no seu local correto.
- Tratamento dos problemas de oclusão e composição de cena.

Enfim, a área se apresenta de forma muito promissora tanto para pesquisa, quanto para a sua imediata aplicação na indústria. A maneira intuitiva de interação credencia esse modelo de interface a se tornar um “padrão” em um futuro próximo, porém muito caminho deve ainda ser percorrido. Espera-se que a passos longos.

Referências

- [1] MILGRAM, P. et al. **Augmented Reality: A Class of Displays on the Reality-Virtuality Continuum**, Telem manipulator and Telepresence Technologies, Vol. 2351, p. 282-292, 1994.
- [2] BOWSKILL, J.; DOWNIE, J. **Extending the Capabilities of the Human Visual System: an Introduction to Enhanced Reality**. SIGGRAPH Computer Graphics, v. 29, n. 2, p. 61-65, 1995.
- [3] AZUMA, R. T. **A Survey of Augmented Reality**. In Presence: Teleoperators and Virtual Environments 6, 4, p. 355-385, 1997.
- [4] MORIE, J. F. **Inspiring the Future: Merging Mass Communication**, Art, Entertainment and Virtual Environments. Computer Graphics. Vol. 28. Nro. 2. pp. 135-138, 1994.
- [5] 5DT. **Fifth Dimension Technologies**. Disponível em: <<http://www.5dt.com/>>. Acesso em: 16 jul. 2007.
- Adams, L. (1994). “Visualização e Realidade Virtual”. Makron Books. p. 255-259.
- [6] WORKBENCH. **The Virtual Reality Responsive Workbench**. Disponível em: <<http://www.ait.nrl.navy.mil/3dvmel/projects/Workbench/Workbench.html>>. Acesso em: 16 jul. 2007.
- [7] CAVE. **Advanced Perception Research Group / CAVE system**. Japan Agency for Marine-Earth Science and Technology. Disponível em: <<http://www.es.jamstec.go.jp/esc/research/Perception/vr.en.html>>. Acesso em 17 jul 2007.
- [8] KRUEGER, W.; FROELICH, B. **The Responsive Workbench**. IEEE Computer Graphics and Applications. 1994.
- [9] CRUZ-NEIRA, C.; SANDIN, D. J.; DEFANTI, T. A. **Surround-screen Projection-based Virtual Reality: The Design and Implementation of the CAVE**. In: ACM SIGGRAPH '93, 1993.
- [10] ZUFFO, J. A. et al. **Caverna Digital - Sistema de Multiprojeção Estereoscópico baseado em Aglomerados de PCs para Aplicações Imersivas em Realidade Virtual**. In: IV SVR - SBC Symposium on Virtual Reality. Anais... pp. 139-147. Florianópolis. 2001.
- [11] NETTO, A. V.; MACHADO, L. S.; OLIVEIRA, M. C. F. **Realidade Virtual - Definições, Dispositivos e Aplicações**. Revista Eletrônica de Iniciação Científica - REIC. Ano II. Vol. II. Número I. ISSN 1519-8219. 2002.
- [12] INSLEY, S. **Augmented Reality: Merging the Virtual and the Real**. Oregon State University., 2003
- [13] GEOGIA TECH. **Dart: The Designer's Augmented Reality Toolkit**. Disponível em: <<http://www.cc.gatech.edu/dart>> Acesso em: 01 mai. 2010.
- [14] STUDIERSTUBE. Disponível em: <<http://studierstube.icg.tu-graz.ac.at/>>. Acesso em: 29 abr. 2010.
- [15] BIMBER, O.; RASKAR, R. **Modern Approaches to Augmented Reality**. In: SIGGRAPH '05: ACM SIGGRAPH 2005 Courses, New York, NY, USA: ACM Press. 2005.
- [16] KIRNER, C.; TORI, R. **Introdução a Realidade Virtual, Realidade Misturada e Hiper-realidade**. In: Realidade Virtual: Conceito e Tendências. Cap. 1. Editora Mania de Livro. 2004.
- [17] VALLINO, J. **Introduction to Augmented Reality**. Disponível em: <<http://www.se.rit.edu/~jrv/>>. Acesso em: 30 abr. 2010.
- [18] BRAZ, J.; PEREIRA, J. **Tarcast: Uma Taxonomia para Sistemas de Realidade Aumentada**. In: Actas do 13o Encontro Português de Computação Gráfica, Vila Real, Portugal, 2005.
- [19] BICHO, A. L. et al. **Programação Gráfica 3D com OpenGL, Open Inventor e Java 3D**. Revista Eletrônica de Iniciação Científica (REIC), Sociedade Brasileira de Computação, V. II, nro. I, ISSN 1519-8219. 2002.
- [20] SHREINER, D. **The OpenGL Reference Manual - The Bluebook**. Disponível em: <<http://www.glprogramming.com/blue/>>. Acesso em: 30 abr. 2010.
- [21] SELMAN, D. **Java 3D Programming**. Manning Publications Co, 2002.
- [22] KATO, H. et al. **ARToolkit PC**. Disponível em < <http://www.hitl.washington.edu/artoolkit/>>. Acesso em 30 abr. 2010.
- [23] HUMAN INTERFACE TECHNOLOGY LAB. **ARToolkit**. Disponível em: <<http://www.hitl.washington.edu/artoolkit/>> Acesso em: 30 abr. 2010.

- [24] HALLER, M. et al. **Combining ARToolkit with Scene Graph Libraries**. In: *Augmented Reality Toolkit, The First IEEE International Workshop*, p. 2, 2002.
- [25] LOOSEA, J. et al. **ARToolkit for Openscenegraph**. Disponível em: <<http://www.artoolworks.com/community/osgart>>. Acesso em: 30 abr. 2010.
- [26] SEMENTILLE, A. C. et al. **A Motion Capture System using Passive Markers**. In: *VRCAI '04: Proceedings of the 2004 ACM SIGGRAPH International Conference on Virtual Reality Continuum and its Applications in Industry*, New York, NY, USA: ACM Press, p. 440_447, 2004.
- [27] GEIGER, C. et al. **Jartoolkit – a Java Binding for ARToolkit**. In: *Augmented Reality Toolkit, The First IEEE International Workshop*, p. 5, 2002.
- [28] GUIMARÃES, M. P. et al. **Ferramentas para o Desenvolvimento de Aplicações de Realidade Virtual e Aumentada**. In: *Realidade Virtual: Conceito, Projeto e Aplicações*. Cap. 6. Editora SBC, p. 108-128, 2007.
- [29] GUIMARÃES M. P., et al. **Montando ambientes de multiprojeção de baixo custo**. Minicurso. Brazilian Symposium on Virtual Reality (SVR 2006). Belém. 2006.
- [30] CARDOSO, A.; LAMOUNIER Jr. E. (2004). **Realidade Virtual na Educação e Treinamento**. In: *Realidade Virtual: Conceito e Tendências*. Cap. 17. Editora Mania de Livro, 2004.
- [31] NUNES, F. L. S. et al. **Aplicações Médicas usando Realidade Virtual e Realidade Aumentada**. In: *Realidade Virtual: Conceito, Projeto e Aplicações*. Cap. 10. Editora SBC, p. 222-255, 2007.
- [32] TORI, R. et al.. **Jogos e Entretenimento com Realidade Virtual e Aumentada**. In: *Realidade Virtual: Conceito, Projeto e Aplicações*. Cap. 10. Editora SBC, p. 192-222, 2007.
- [33] CARDOSO, A. et al. **Conceitos de Realidade Virtual e Aumentada**. In: *Tecnologias para o Desenvolvimento de Sistemas de Realidade Virtual e Aumentada*. Cap. 1. Editora Universitária da UFPE, p. 1-16, 2007.
- [34] KIYOKAWA, K., KURATA, Y., OHNO, H. **An optical see-through display for mutual occlusion of real and virtual environments**. *Proceedings of the IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)*. 2000.
- [35] BILLINGHURST, M., KATO, H., POUPYREV, I., **Tangible augmented reality**. *SIGGRAPH Asia '08: ACM SIGGRAPH ASIA 2008 courses*. pp 1-10. 2008.
- [36] FARBIZ, F. et al. **A multimodal augmented reality DJ music system**. *6th International Conference on Information, Communications Signal Processing*. pp. 1-5. 2007
- [37] KATO, H.; BILLINGHURST, M. **Marker Tracking and HMD Calibration for a Video-Based Augmented Reality Conferencing System**. *IWAR '99: Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality*. 1999.
- [38] FIALA, M. **ARTag, a Fiducial Marker System Using Digital Techniques**. *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, v. 2, pp. 590 – 596, 2005.
- [39] SANCHES, S. R. R., et al. **The Generation of Scenes in Mixed Reality Environments using the Chromakey Technique**. *ICAT '07: Proceedings of the 17th International Conference on Artificial Reality and Telexistence*. pp. 296-297. 2007.
- [40] PANG, Y., et. al. **A markerless registration method for augmented reality based on affine properties**. *AUIC '06: Proceedings of the 7th Australasian User interface conference*. Pp. 25 – 32. 2006.
- [41] SCHALL, G., et al. **Global pose estimation using multi-sensor fusion for outdoor Augmented Reality**. *8th IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2009*, 2009.
- [42] PIEKARSKI, W.; THOMAS, B., **ARQuake: the outdoor augmented reality gaming system**. *ACM Commun.* V 45. n.1, pp. 36-38. 2002.
- [43] GREENLEAF, W. J. **Neuro/Orthopedic Rehabilitation and Disability Solutions Using Virtual Reality Technology**. AKAY, M.; MARSH, A. *Information Technologies in Medicine, Volume II: Rehabilitation and Treatment*. John Wiley & Sons, Inc. cap 1, p. 3-18. 2001.
- [44] UMLAUF, E. J., et al. **ARLib: The Augmented Library**. *Proceedings of the First IEEE International Workshop on ARToolKit*. 2002.
- [45] SCHMALSTIEG, D.; et al., W. **Studierstube Augmented Reality Project**. *Presence*, Vol. 11, No. 1, February 2002, p33–54.
- [46] ELOP, **Elbit Systems Electro-Optics** ELOP Ltd. Disponível em: <http://www.el-op.com/category/head-up_displays.html> Acesso em: 26 de julho de 2010.