

UNIVERSIDADE DO EXTREMO SUL CATARINENSE – UNESC
CURSO DE CIÊNCIA DA COMPUTAÇÃO

RAFAEL LAZZARI

REALISMO VISUAL: RENDERIZAÇÃO DE IMAGENS EM JOGOS
COMPUTACIONAIS

CRICIÚMA

2012

RAFAEL LAZZARI

**REALISMO VISUAL: RENDERIZAÇÃO DE IMAGENS EM JOGOS
COMPUTACIONAIS**

Trabalho de Conclusão de Curso, apresentado para obtenção do grau de Bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense.

Orientador: Prof^a. Msc. Leila Laís Gonçalves

**CRICIÚMA
2012**

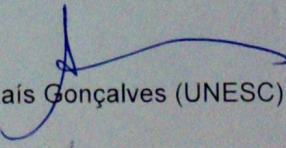
RAFAEL LAZZARI

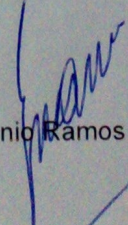
**REALISMO VISUAL: RENDERIZAÇÃO DE IMAGENS EM JOGOS
COMPUTACIONAIS**

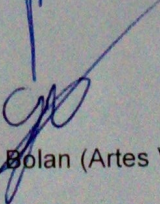
Trabalho de Conclusão de Curso aprovado pela Banca Examinadora para obtenção do Grau de Bacharel, no Curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC, com Linha de Pesquisa em Computação Gráfica.

Criciúma, 26 de junho de 2012.

BANCA EXAMINADORA


Profa. MSc. Leila Laís Gonçalves (UNESC) - Orientadora


Prof. M.Eng. Evânio Ramos Nicoleit (UNESC)


Prof. Esp. Giacomo Bolan (Artes Visuais - UNESC)

À meus pais, que sempre são
exemplos de grandes guerreiros,
à minha irmã que tanto incomodo e
à minha namorada Graciela,
que sempre está ao meu lado.

AGRADECIMENTOS

Agradeço a Deus por permitir que eu conquistasse tudo o que tenho hoje e a terminar mais esta fase da minha vida.

Obrigado a minha família por sempre me apoiar nas minhas decisões, ao meu pai Ivo Lazzari que me apoiou incondicionalmente nos estudos e sustento, à minha mãe Eliane L. A. Lazzari por me apoiar e me cuidar sempre que me era necessário e à minha querida irmã Carolina L. Lazzari pelos momentos juntos.

Agradeço a minha namorada Graciela Verdum por ter me ajudado nesta fase e por sempre ser um doce para mim. Obrigado também pelo apoio da família Verdum.

Agradeço a minha orientadora Mestra Leila Laís Gonçalves por me indicar o melhor caminho para a conclusão deste trabalho.

A UNESC, Universidade do Extremo Sul Catarinense, por acreditar e incentivar os grupos de pesquisas e pelo apoio institucional.

Agradeço a todos que contribuíram com alguma coisa para que eu chegasse até onde estou hoje.

“Para o triunfo do mal só é preciso que
os bons homens não façam nada.”

(Edmund Burke)

RESUMO

A construção de modelos para geração de imagens sintéticas que ofereçam Realismo Visual é um dos principais objetivos da Computação Gráfica, pois essa característica de fidelidade com o mundo real está relacionada e observada a uma boa qualidade, que influencia na escolha e atrai os usuários adeptos a essa área. Logo, tornou-se fundamental a inclusão dessa característica, que é facilmente percebida em aplicações como simulação, cenários e personagens de filmes e jogos, realidade virtual, criação de ambientes virtuais e reconstruções realísticas tridimensionais. Uma das preocupações do segmento de jogos computacionais é a grande demanda de processamento necessária à renderização de imagens hiper-realistas. Neste trabalho apresentam-se os processos, as fases de renderização, assim como as técnicas utilizadas para obtenção do realismo visual, de maneira a explicitar o motivo de ele ser uma tendência na computação gráfica atual e provavelmente se tornar algo permanente. Expõem-se vários tipos de jogos computacionais, que são conceituados por vários autores e citam-se vários títulos de games conhecidos e famosos, que também colaboram para demonstrar o quanto o realismo nos jogos eletrônicos evoluíram. Apresenta-se as game engines, as engines gráficas, o que são e quais as principais, que são o OpenGL e o DirectX, e outros engines derivados desses principais, todos são conceituados e são mostrados as características e diferenças de ambos, onde são aplicadas e para qual motivo foram desenvolvidos. Esses conceitos são vistos para obtenção de avaliação de técnicas de renderização de imagens em jogos computacionais, por meio da identificação e contextualização dos atributos que compõem o realismo, assim como estudos em ambientes de desenvolvimento 2D/3D e dos resultados adquiridos no protótipo gerado com a ferramenta Blender, para essa avaliação.

Palavras-chave: Realismo Visual; Computação Gráfica; Jogos Computacionais; Imagens Digitais; 2D; 3D; Blender.

ABSTRACT

The construction of models for generating synthetic images that offer Visual Realism is one of the main goals of computer graphics, because this characteristic of fidelity to the real world is observed and related to good quality, which influence the choice and attracts fans to users this area. Soon, it became essential to include this feature, which is easily realized in applications such as simulation, scenarios and characters from movies and games, virtual reality, virtual environments and creating realistic three-dimensional reconstructions. One of the concerns of the gaming industry is the large computational demands of processing required to render hyper-realistic images. This paper presents the processes, stages of rendering, as well as the techniques used to obtain the visual realism in order to explain the reason he is a current trend in computer graphics and probably become permanent. We describe the various types of computer games, which are highly regarded by several authors and it cites several titles of games known and famous, who also collaborate to show how much realism in video games have evolved. It presents the game engines, the graphics engines, what are they and what are the main, they are DirectX and OpenGL, and other derivatives of these main engines, all are respected and are shown the characteristics and differences of both, which are applied and for which reason they were developed. These concepts are seen to obtain evaluation techniques rendering images in computer games, and by identifying contextual attributes that make up the realism, as well as studies in development environments 2D/3D and the results obtained with the prototype generated Blender tool for this evaluation.

Key-words: Visual Realism, Computer Graphics, Computer Games, Digital Imaging, 2D, 3D; Blender.

LISTA DE ILUSTRAÇÕES

Figura 1 - Os quatro universos na abstração de uma esfera.....	21
Figura 2 - Etapas da imagem digital.....	22
Figura 3 - Imagem 2D.....	24
Figura 4 - Imagem digital 2D.....	24
Figura 5 - Imagem 3D.....	25
Figura 6 - Imagem digital 3D.....	27
Figura 7 - Cenário digital 2D.....	28
Figura 8 - Cenário digital 3D.....	29
Figura 9 – Textura.....	30
Figura 11 - Reticulado uniforme da representação matricial bidimensional no plano.....	31
Figura 12 - Representação vetorial e matricial de uma elipse.....	32
Figura 13 - Complexidade do mundo real: cores, formatos e texturas.....	33
Figura 14 - Formas primitivas.....	36
Figura 15 - Modelagem de forma livre.....	36
Figura 16 - Geometria Sólida Construtiva.....	37
Figura 18 - Secção transversal.....	38
Figura 19 – Revolução.....	38
Figura 20 - Rasterização de uma reta. Em B, é incluído um tratamento de anti-aliasing.....	39
Figura 21 - Directional light.....	41
Figura 22 - Point light	42
Figura 23 - Spot light.....	42
Figura 24 - Texturas	43
Figura 25 - UVW.....	44
Figura 26 - Textura maçã.....	44
Figura 27 - Mapa de reflexão.....	45
Figura 28 - Bump Map.....	46
Figura 29 - Ruídos.....	46
Figura 30 - Utilização do mip map.....	47
Figura 31 - Fonte de luz aplicada na textura.....	47

Figura 32 - Taças de vinho.....	48
Figura 33 - Projected shadows.....	49
Figura 34 - Jogo 2D.....	51
Figura 35 - Jogo 2D.....	52
Figura 36 – Jogo 3D.....	53
Figura 37 - Zelda 2D.....	55
Figura 38 – Zelda 3D.....	55
Figura 39 - Wolfenstein 3D.....	57
Figura 40 – Doom.....	57
Figura 41 – Silent Hill.....	58
Figura 42 – Medal of Honor.....	59
Figura 43 – Heavenly Sword.....	59
Figura 44 – Tatsunoko vs. Capcom: Ultimate All-Stars – 2009.....	60
Figura 45 - Marvel vs Capcom 3: Fate of Two Worlds – 2011.....	60
Figura 46 - Gorilla - Everett Gunther.....	71
Figura 47 - Mulher com Chapéu - Zoltan Miklosi.....	71
Figura 48 - Registe do Juggernaut - Derek Watts.....	71
Figura 49 - Yofrankie.....	72
Figura 50 – Textura madeira.....	73
Figura 51 – Textura cesta.....	73
Figura 52 – Textura laranja.....	73
Figura 53 - Modelos geométricos.....	74
Figura 54 - Texturização.....	74
Figura 55 - Aplicação de Bump map.....	75
Figura 56 - Iluminação.....	75
Figura 57 - Finalização.....	76
Figura 58 - Finalização, de cima e 3D.....	76

LISTA DE TABELAS

Tabela 1 - Quadro comparativo 2D e 3D.....	28
Tabela 2 – Análise do protótipo.....	78

LISTA DE ABREVIATURAS E SIGLAS

2D	Duas Dimensões
3D	Três Dimensões
API	Application Program Interface
BGE	Blender Game Engine
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CG	Computação Gráfica
HAL	Hardware Abstraction Layer
IA	Inteligência Artificial
PC	Personal Computer
UFRGS	Universidade Federal do Rio Grande do Sul

SUMÁRIO

1 INTRODUÇÃO.....	15
1.1 OBJETIVO GERAL.....	16
1.2 OBJETIVOS ESPECÍFICOS.....	16
1.3 JUSTIFICATIVA.....	16
1.4 ESTRUTURA DO TRABALHO.....	17
2 COMPUTAÇÃO GRÁFICA E REALISMO VISUAL EM IMAGENS 2D E 3D.....	19
2.1 IMAGEM DIGITAL (ID).....	20
2.2 REPRESENTAÇÃO ESPACIAL: IMAGEM 2D E 3D.....	23
2.2.1 Imagem bidimensional (2D).....	23
2.2.2 A imagem tridimensional (3D).....	25
2.2.3 Imagem 2D versus 3D.....	27
2.3 IMAGENS VETORIAIS E BITMAP.....	30
2.4 TÉCNICAS DE COMPUTAÇÃO GRÁFICA E REALISMO VISUAL.....	32
2.4.1 Realismo e o Modelo.....	35
2.4.2 Realismo e a Aparência.....	40
3 JOGOS DIGITAIS.....	50
3.1 CONTEXTOS E CARACTERÍSTICAS DE JD.....	50
3.2 EVOLUÇÃO DO REALISMO NOS GRÁFICOS.....	53
3.3 GAME ENGINES.....	61
3.3.1 Engines Gráficas.....	61
3.3.1.1 OPENGL.....	61
3.3.1.2 DIRECTX.....	62
3.3.1.3 OUTRAS.....	63
4 TRABALHOS CORRELATOS.....	64
4.1 ESTUDOS DE TÉCNICAS REALISTAS DE RENDERIZAÇÃO EM TEMPO REAL.....	64
4.2 3D BUILDER: UMA GAME ENGINE PARA A CRIAÇÃO DE JOGOS 3D.....	65
4.3 PROJETO E DESENVOLVIMENTO DE JOGOS COMPUTACIONAIS.....	65
4.4 OS CONCEITOS ESTÉTICOS-VISUAIS DOS JOGOS DIGITAIS.....	66
4.5 PROGRAMAÇÃO DE JOGOS 2D USANDO O MRDX E FUNDAMENTOS DE	

JOGOS 3D.....	67
5 REALISMO VISUAL: RENDERIZAÇÃO DE IMAGENS EM JOGOS COMPUTACIONAIS.....	68
5.1 METODOLOGIA.....	68
5.1.1 Levantamento Bibliográfico.....	69
5.1.2 Escolha da ferramenta utilizada para confecção de modelos	69
5.1.3 Prototipação: realismo visual e textura.....	72
5.1.4 Análise do protótipo.....	78
5.2 DISCUSSÃO E ANÁLISE DOS RESULTADOS OBTIDOS.....	78
6 CONCLUSÃO.....	80
REFERÊNCIAS.....	81

1 INTRODUÇÃO

Um dos principais objetivos da Computação Gráfica (CG) é construir modelos para geração de imagens sintéticas realistas. O Realismo Visual, em CG, pode ser considerado em duas etapas: a estática, foco do TCC presente, na qual são tratados os objetos e cenas estáticas buscando o realismo fotográfico ou fotorrealismo, e a dinâmica que está relacionada ao movimento da cena e seus personagens focando a animação computacional.

O Realismo Visual é sinônimo de boa qualidade e é fundamental e perceptível em aplicações como simulação, cenários e personagens de filmes e jogos, realidade virtual, criação de ambientes virtuais e reconstruções realísticas tridimensionais. Para obtê-lo, fazem-se modelos físicos cada vez mais sofisticados, que vêm sendo utilizados para modelar as interações das fontes de luz com os objetos. Outras evoluções podem ser observadas nas formas de representação geométrica (bi e tridimensional - 2D e 3D) e apresentação dos materiais que caracterizam uma imagem. Quanto mais sofisticado o modelo e as representações, mais demanda há de recursos tanto de software quanto hardware para síntese e renderização dessas imagens.

O realismo visual em jogos, observada na qualidade dos gráficos apresentados, é um dos requisitos fundamentais e indicador forte de qualidade que influencia na escolha e opinião dos usuários. Que grau de realismo será possível atingir em jogo computacional? Ainda que não esteja disponível a imersão total, na qual não há limite de interatividade e diferença entre o real e virtual, muito já se avançou nesse sentido. Porém, uma das preocupações do segmento de jogos de computador é a grande demanda de processamento necessária à renderização de imagens hiper realistas. Este é considerado um dos gargalos da indústria de jogos.

Quanto de realismo visual tem sido inserido nos jogos? Qual a relação de custo e benefício entre o real ideal e o virtual possível e quanto isso tem onerado no processamento, na síntese e renderização dessas imagens?

Para responder estas e outras questões relacionadas ao realismo visual e desempenho no processamento, na síntese e renderização de imagens em jogos de computador torna-se necessário analisar os recursos envolvidos no seu desenvolvimento e utilização. Neste sentido, este trabalho visa avaliar técnicas de computação gráfica para síntese e renderização de imagens 2D e 3D em jogos computacionais focando na texturização, na qualidade do realismo visual e no custo computacional.

1.1 OBJETIVO GERAL

Avaliar técnicas de texturização na renderização de imagens para jogos computacionais evidenciando a qualidade do realismo visual.

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos deste trabalho são:

- a) identificar os atributos que compõem o realismo visual em 3D a partir da sua geometria e das técnicas computacionais;
- b) contextualizar a evolução do realismo visual;
- c) comparar técnicas de computação gráfica para síntese e renderização de imagens 2D e 3D em jogos computacionais evidenciando a qualidade do realismo visual;
- d) desenvolver protótipos para validação das técnicas;
- e) analisar e avaliar os resultados gerados nos protótipos.

1.3 JUSTIFICATIVA

Alguns ambientes gráficos clássicos, como jogos virtuais de computadores, estão se tornando cada vez mais imersivos, dando mais possibilidades para o usuário interagir e mudar o meio em que está imerso virtualmente (ANDRADE, 2007).

As técnicas de realismo possuem um grande número de operações matemáticas que devem ser processadas. Em um jogo computacional são executadas ações em tempo real, necessitando que estes cálculos sejam executados em pouquíssimo tempo, tornando determinados níveis de realismo totalmente inviáveis para serem processados pelos computadores, sem muitos recursos, dos usuários finais. São necessárias otimizações nos algoritmos de processamento gráfico para que estes usuários possam usufruir destes novos recursos computacionais. Um exemplo de otimização é a redução de polígonos dos objetos 3D, para que uma placa gráfica menos potente possa processar sem demora.

O realismo visual é um dos atributos, quando se refere a uma boa qualidade e é fundamental e percebível em aplicações como simulação, cenários e personagens de filmes e jogos, realidade virtual, criação de ambientes virtuais e reconstruções realísticas tridimensionais. Podem-se considerar duas etapas para a sua aplicação em imagens, a estática

e a dinâmica. A etapa dinâmica visa o movimento de cenas e personagens contemplados na animação computacional. Na etapa estática, são estudadas e descritas as técnicas e teorias responsáveis por gerar objetos e cenas estáticas com realismo fotográfico ou fotorrealismo. É nessa última etapa que se aprofunda este Trabalho de Conclusão de Curso (TCC).

Para a execução destas tarefas citadas são necessárias análises das técnicas de realismo, estudos em ambientes de desenvolvimento 2D/3D e um relatório comparativo das técnicas e dos resultados gerados nos protótipos de exemplo.

1.4 ESTRUTURA DO TRABALHO

No Capítulo 1 apresenta-se a Introdução do presente trabalho, ela remete aos assuntos que serão descritos no desenvolvimento. Nela se apresentam os objetivos, geral e específicos, assim como as justificativas para esse relatório.

O Capítulo 2 explana sobre Computação Gráfica e Realismo Visual em Imagens 2D e 3D, defini-se o que são imagens de duas e três dimensões assim como a percepção de espacialidade que as mesma ocupam, faz-se também um comparativo desses dois tipos de figuras. Neste capítulo se apresenta os processos, as fases de renderização, assim como as técnicas utilizadas para obtenção do realismo visual, e se explica o motivo dele ser uma tendência na computação gráfica atual e provavelmente se tornar algo permanente.

No Capítulo 3 apresentam-se conceitos, tipos, exemplares dos jogos computacionais, são expostos conceitos de diversos autores e citam-se vários títulos de games conhecidos e famosos, que também colaboram para demonstrar o quanto o realismo nos jogos eletrônicos evoluíram. Apresenta-se as *game engines*, as *engines* gráficas, o que são e quais as principais, que são o OpenGL e o DirectX, e outros engines derivados desses principais, todos são conceituados e são mostrados as características e diferenças de ambos, onde são aplicadas e para qual motivo foram desenvolvidos. Nesse capítulo, também busca-se descrever o porquê do realismo nos jogos ter chegado para ficar de vez.

O Capítulo 4 mostra os trabalhos correlatos, ou seja, relacionados ao trabalho presente, *REALISMO VISUAL: RENDERIZAÇÃO DE IMAGENS EM JOGOS COMPUTACIONAIS*, essa relação é demonstrada, a partir das sucintas descrições desses trabalhos, o que permite, que se observe o que há de análogo entre eles.

No Capítulo 5, apresenta-se o trabalho desenvolvido, o protótipo, o método de confecções e os recursos utilizados. Faz-se também análises sobre o modelo e se sugere novas pesquisas para complementação.

E ao final a Conclusão, onde se resume o que foi obtido a partir dos estudos, pesquisas e práticas, assim como também se expõe idéias para melhorias, se descreve o que funcionou, ou não; e se sugere futuras discussões correlatas a este TCC.

2 COMPUTAÇÃO GRÁFICA E REALISMO VISUAL EM IMAGENS 2D E 3D

De acordo com a *International Standards Organization (ISO)*, a Computação Gráfica (CG) é a área da Ciência da Computação que estuda a geração, manipulação e interpretação de modelos e imagens de objetos incluindo técnicas e métodos computacionais que convertem dados para dispositivos gráficos e vice-versa. Neste contexto, As imagens constituem uma representação visual de objetos de duas ou três dimensões descritos por meio de especificações abstratas. De acordo com Gomes e Velho (1998), a imagem é a materialização de grande parte dos processos da CG sendo que está presente em todas as suas áreas, seja como produto final (visualização), ou como parte do processo de interação (modelagem). A CG engloba assim, a síntese, o processamento e a análise de imagens (PERSIANO, OLIVEIRA, 1989).

A Síntese de Imagens de objetos reais ou imaginários foca nas representações visuais de objetos sintetizados computacionalmente considerando as especificações geométricas e visuais de seus componentes a partir de modelos computacionais. Os modelos são criados considerando a descrição de objetos em suas formas geométricas básicas como segmento de reta, polígonos, poliedros, esfera, etc. A síntese pode ser identificada como Visualização Científica ou Computacional cujo objetivo é a representação gráfica da informação.

O Processamento de Imagens considera a forma digital e suas transformações buscando melhorar ou realçar suas características visuais. O processamento realiza a análise de cenas ou a reconstrução de modelos de objetos 2D ou 3D a partir de suas imagens prontas. Já a Análise de Imagens considera as imagens digitais e as analisa para obtenção de características desejadas (por exemplo, a especificação dos componentes de uma imagem a partir de sua representação visual).

Na síntese de imagens, existe uma complexa relação entre a realidade e o objeto sintetizado sendo possível observar que este processo busca uma relação direta com o mundo real. Conforme Couchot (1996), a imagem numérica não representa mais o mundo real, ela o simula. Ela o reconstrói, fragmento por fragmento, propondo dele uma visualização numérica que não mantém mais nenhuma relação direta com o real, nem física, nem energética. A imagem digital (ID) não é mais o ponto de chegada visual de um corte ou de um enquadramento óptico que manifesta por projeção - na ordem da representação - uma essência objetiva atribuída por antecipação ao mundo e revelada pelo olhar de um sujeito universal e soberano. Não é mais a passagem do fundo à superfície, mas um acontecimento aleatório,

ponto de chegada de um processo, que remete ao jogo de toda uma série de mediações específicas que o traduzem e o conduzem até o estágio de imagem terminal (RÉNAUD, 1989, tradução nossa).

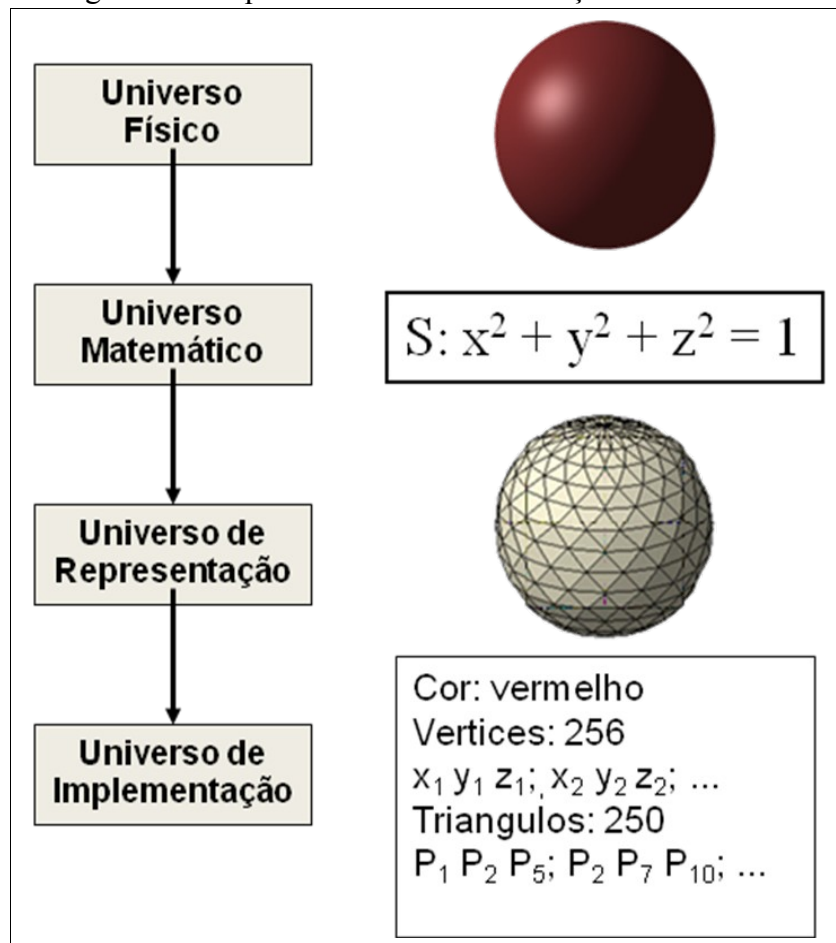
De acordo com sua representação visual e composição as imagens digitais caracterizam-se por: suas dimensões na representação espacial (2D e 3D) e por seus elementos de composição (vetores ou mapas de bits). De acordo com sua representação espacial as IDs podem ser chamadas de imagens bidimensionais (2D) e imagens tridimensionais (3D). A modelagem geométrica envolve a forma de representação dos objetos, técnicas de modelagem, modificadores e interface de modelagem. Quanto à sua composição as IDs podem ser classificadas como imagens vetoriais ou imagens de mapas de bits (bitmap, raster, de varredura ou pixel maps - mapa de pixels).

2.1 IMAGEM DIGITAL (ID)

O termo “imagem” deriva de três vertentes: do latim *imago*, significa representação visual de um objeto; do grego antigo *eidos*, raiz etimológica do termo *idea* ou *eidea*, conceito do idealismo de Platão que considerava a ideia da coisa (a sua imagem) como uma projeção da mente; e do realismo de Aristóteles que julgava a imagem como uma representação mental de um objeto real. Nas ciências exatas, como na Matemática, o termo "imagem" é entendido como representação de um objeto / objeto especializado, que para tal, exige técnicas e ferramentas especiais.

De acordo com Azevedo e Conci (2003), um objeto gráfico é representado por meio da especificação de seus atributos: geométricos (estabelecem as propriedades métricas dos objetos); topológicos (tratam da forma dos objetos), de cor (definem a informação de textura dos objetos). Para representar o mundo real no mundo virtual, usa-se na CG o paradigma de abstração: de níveis (contínuo e discreto) ou dos quatro universos (Físico, Matemático, Representação, Implementação) considerando os diferentes níveis e processos na representação de uma imagem na visão computação ilustrado na Figura 1 abaixo.

Figura 1 - Os quatro universos na abstração de uma esfera



Fonte: Toledo (2011)

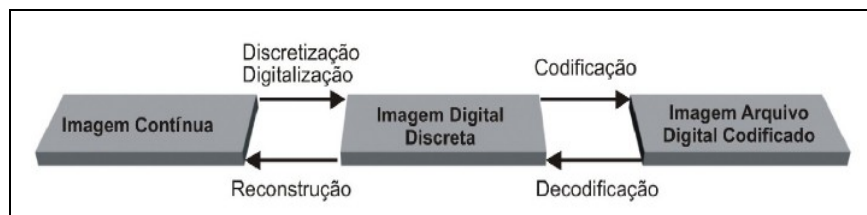
De acordo com Toledo (2011), no Universo Físico, as imagens reais possuem um número ilimitado de cores ou tons. Os diversos atributos do objeto gráfico são descritos por meio de modelos definidos em espaços topológicos utilizando funções contínuas (Universo Matemático) ou de classe de diferenciabilidade mais alta. No Universo Matemático temos o uso de Funções no Espaço de Cor. Assim, a imagem refere-se à função bidimensional de intensidade da luz $f(x,y)$, onde x e y denotam as coordenadas espaciais e o valor f em qualquer ponto (x, y) é proporcional ao brilho (ou níveis de cinza) da imagem naquele ponto. Uma imagem pode ser definida por

$f: U \subset \mathbb{R}^2 \rightarrow C$, onde:

- U é um conjunto chamado suporte da imagem. O conjunto de valores de f , que é um subconjunto de C , é chamado de conjunto de cores da imagem;
- $C = \mathbb{R}^n$ é um espaço de cor (conjunto de cores). Se n for 3, temos um espaço de representação de cores tricromático, em geral um espaço com base de primárias $R(\text{red})$, $G(\text{green})$ e $B(\text{blue})$.

No Universo de Representação do domínio discreto, os diversos atributos do objeto gráfico são discretizados com o objetivo de se obter uma representação finita, que possa ser utilizada no computador. O processo de aquisição de uma imagem digital envolve amostragem que indica o número de pontos amostrados de uma imagem digitalizada – resolução; e quantização indicando a quantidade de níveis de tons que pode ser atribuído a cada ponto digitalizado – profundidade de cor. Como procedimentos na aquisição de imagem digital têm-se (AZEVEDO; CONCI, 2003): a discretização (conversão da imagem contínua em uma representação discreta; codificação (a partir da representação discreta da imagem são gerados um conjunto de dados representativos da imagem que podem ser transformados no formato de arquivos); reconstrução (processo inverso da discretização); decodificação (processo oposto à codificação no qual acessam-se informações codificadas na forma de uma representação discreta). A Figura 2 representa as etapas e procedimentos da aquisição e representação da imagem digital.

Figura 2 - Etapas da imagem digital



Fonte: Azevedo e Conci (2003)

Uma imagem digital é uma imagem $f(x, y)$ discretizada tanto em coordenadas espaciais quanto em brilho. Uma imagem digital pode ser descrita como sendo uma matriz $(N \times M)$ cujos índices de linhas e de colunas identificam um ponto na imagem, Os elementos dessa matriz digital são chamados de elementos da imagem, elementos da figura, "*pixels*" (abreviatura de "*picture elements*" - elementos de figura). Os valores de um pixel $(p(x,y))$ são inteiros positivos e indicam a intensidade de cor em cada posição $[x,y]$ da imagem. Na digitalização a imagem recebe um tamanho adimensional, em pixels. A razão entre o número de pixels e o espaço que a contém define a qualidade de resolução especial de uma imagem que em geral é medida em pontos por polegada ou dpi (*dots per inch*). Quanto mais pixels uma imagem tiver melhor é a sua resolução e qualidade.

Quanto ao Universo de Implementação têm-se a decisão quanto à escolha de Codificação e Formatos. Neste sentido estão disponíveis tecnologias, APIs, bibliotecas, formatos diferenciados de arquivos dentre outras possibilidades. Imagens digitais são úteis

quando estão armazenadas em uma forma que possa ser utilizada por outras aplicações (FACON 1993, p. 28). Logo, para que as imagens possam ser manipuladas, se faz necessário que as armazene de forma padronizada.

2.2 REPRESENTAÇÃO ESPACIAL: IMAGEM 2D E 3D

A percepção de “espacialidade” de uma imagem pode ser vista como a capacidade que o ser humano tem de distinguir a forma, as cores, a textura e a relação espacial existente entre os objetos. O mundo real é formado por um espaço fisicamente tridimensional sendo que a dimensão é um parâmetro ou medida requerida para definir as características de um objeto (por exemplo, largura, altura, profundidade ou tamanho e forma). Na matemática, a dimensão de um espaço é o número de parâmetros necessários para identificar um ponto e descrever sua posição ou objeto no espaço conceitual.

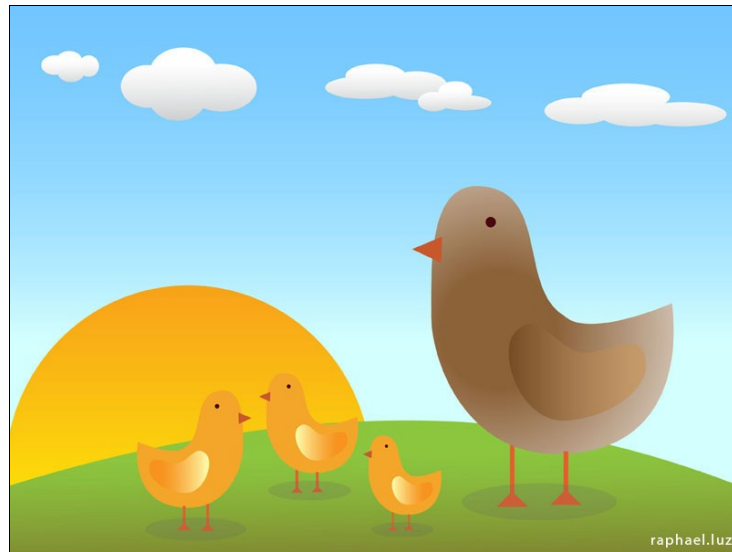
Para a representação gráfica de um objeto bidimensional (2D) são utilizadas duas dimensões: altura e largura. Já nos objetos e imagens tridimensionais (3D) a terceira dimensão, a profundidade, é acrescentada com o uso de técnicas de computação gráfica, entre outros recursos, para dar a impressão de apresentar volume, o que dá maior semelhança com o objeto representado. O estudo da espacialidade busca entender a forma como se percebe a profundidade em imagens bidimensionais e serve tanto para evitar erros na elaboração de imagens, como para possibilitar uma melhor interação com objetos em ambientes virtuais.

2.2.1 Imagem bidimensional (2D)

As imagens bidimensionais (2D) são aquelas que possuem duas dimensões, largura e altura (x,y) e se caracterizam por serem sempre planas, conforme a Figura 3 ilustra. Todas as pinturas, gravuras e fotografias se apresentam primeiramente como bidimensionais. Aumont (2000) descreveu as imagens 2D:

Há para um olho fixo e único, três fontes potenciais de informação sobre a condição plana da imagem; o quadro e o suporte dessa imagem; a superfície (com textura) da própria imagem; os defeitos da representação analógica – em particular, o fato de que, as cores costumam ser menos saturadas e os contrastes menos marcados na imagem do que na realidade (AUMONT, 2000, p. 63).

Figura 3 - Imagem 2D



Fonte: LUZ

Segundo Gonzalez e Woods (2000), uma imagem digital 2D (duas dimensões) é uma imagem $f(x,y)$ discretizada tanto em coordenadas espaciais como em brilho. Uma imagem digital pode ser considerada como sendo uma matriz cujos índices de linha e colunas identificam um ponto na imagem e o correspondente valor do elemento da matriz identifica o atributo da cor naquele ponto.

Para confeccionar imagens bidimensionais há uma ampla quantidade de ferramentas para desenvolver a parte gráfica pretendida. A Figura 4 a seguir demonstra uma imagem digital 2D:

Figura 4 - Imagem digital 2D

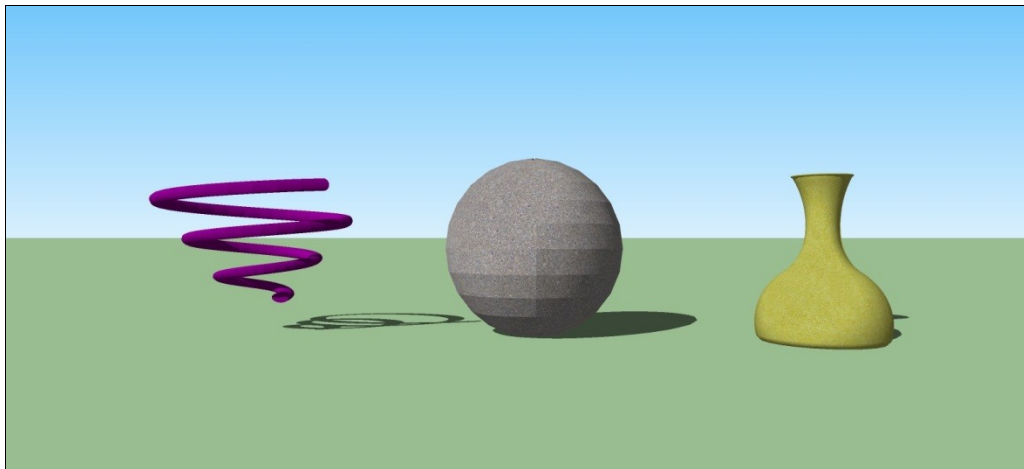


Fonte: MEGAMAN

2.2.2 A imagem tridimensional (3D)

As imagens tridimensionais são aquelas que possuem três dimensões: altura, largura e profundidade (x,y,z). São formadas por polígonos, são gravuras sólidas, não planas. Como a Figura 5 ilustra uma imagem tridimensional possui volumetria e profundidade, a percepção dessa espacialidade pode ser vista como a habilidade que os seres humanos têm de perceber e diferenciar os formatos, as texturas, as cores e a relação que elas têm com o mundo real.

Figura 5 - Imagem 3D



Fonte: Do autor (2012)

Atualmente tem-se utilizado largamente - para se produzir jogos, efeitos especiais, imagens, em aplicações profissionais, mas também para fins de entretenimento - gráficos tridimensionais.

Segundo Stuart (1996), há três categorias de estímulos visuais usados pelo cérebro para formar uma imagem 3D: dados monoculares (também chamadas de informações estáticas de profundidade, são próprias à imagem formada na retina), informações óculo motoras e elementos estereoscópicas.

O autor identifica como informações monoculares a noção de perspectiva linear, o conhecimento prévio do objeto, a oclusão, a densidade das texturas, a variação da reflexão da luz e as sombras. A noção de perspectiva linear, recurso usado para expressar cenas tridimensionais em superfícies planas, é resultado da aparente diminuição dos tamanhos e das distâncias entre os objetos, à medida que o observador se distancia deles. O conhecimento prévio do tamanho de um objeto é utilizado tanto para determinar sua distância absoluta a partir do observador, quanto às distâncias relativas entre os objetos. A noção de tamanho serve para determinar qual dos objetos está mais próximo ou mais distante. Quanto à oclusão,

interposição ou interrupção de contorno, é responsável pela informação da posição relativa dos objetos causando obstrução da visão de um objeto por outro que está mais próximo do observador. A densidade ou gradiente das texturas é um fenômeno visual baseado no fato de que muitos objetos possuem em sua aparência algum tipo de padrão regular e auxiliam também na percepção do movimento. A variação da reflexão da luz apresentada pela mudança na sua intensidade refletida ao longo de uma superfície de um objeto fornece informações sobre a forma e a curvatura da superfície desse objeto. As sombras e sombreamentos são efeitos úteis na determinação da posição de um objeto em relação a um piso colocado abaixo deste, ou na definição relativa entre objetos.

Para Stuart (1996), as informações visuais óculo motoras são fornecidas pelo movimento dos olhos, produzidos pelos dois conjuntos de músculos do globo ocular gerando a acomodação e a convergência. Na acomodação, os músculos dos olhos relaxam ou contraem para mudar o formato do cristalino (lentes dos olhos) para alterar o foco dos objetos projetados na retina em função da distância desses objetos do observador. Já a convergência considera o grau de rotação dos olhos ao longo do eixo de visão (quando um objeto é focado) para obter informações a respeito da posição e da distância.

Já as informações visuais estereoscópicas pressupõem que os olhos estão posicionados em lugares diferentes, cada olho vê uma imagem de forma diferente. Essa diferença é chamada disparidade binocular. O cérebro usa essas diferenças para obter a distância relativa dos objetos. Atualmente tem-se utilizado largamente - para se produzir jogos, efeitos especiais, imagens, em aplicações profissionais, mas também para fins de entretenimento - gráficos tridimensionais.

Uma imagem digital 3D é formada por milhares de polígonos, formas geométricas como quadrados, triângulos, retângulos, círculos etc. É importante salientar que quanto mais polígonos, maior é o nível de detalhes da imagem, ou seja, aparência de ser mais realista. Cada um desses polígonos possuem uma determinada posição na imagem, assim como dimensão, cor ou textura específica. A Figura 6 a seguir demonstra uma imagem digital 3D:

Figura 6 - Imagem digital 3D



Fonte: CECIL

2.2.3 Imagem 2D versus 3D

A principal diferença que as imagens 2D e 3D apresentam se dá a partir da percepção visual que se tem de cada uma delas, a imagem 3D estaria relacionada à experiência de mundo real, com seus volumes e perspectivas, enquanto que a imagem 2D em sua base é mais distante desse realismo, pois ela não proporciona essa sensação, sendo algo plano, facilmente identificada como uma gravura, uma pintura ou uma cópia.

Ambas têm relação com a realidade, porém as imagens bidimensionais representam a realidade e as tridimensionais, geram uma percepção de ser a realidade, de terem a possibilidade de se fundirem nela, não apenas de ser um desenho. Tratando-se figuras 2D, a fotografia, o cinema (2D) e o vídeo propiciam essa experiência, por se tratar de linguagens que captam a realidade tal qual ela se apresenta. A Figura 7 representa uma floresta em 2D, enquanto a Figura 8 representa uma ilha em 3D. É facilmente perceptível que um ambiente 3D contém mais detalhes, tornando-o mais realista.

Tabela 1 - Quadro comparativo 2D e 3D
QUADRO COMPARATIVO 2D E 3D

Itens	2D	3D
Coordenadas	XY	XYZ
Dimensões	Largura e comprimento	Largura, comprimento e profundidade
Geometria	Plana	Volumétrica
Percepção visual	Representação da realidade	Experiência de mundo real
Complexidade	Menor	Maior
Transformações	rotação, translação, escala, cisalhamento e espelhamento.	rotação, translação, escala, cisalhamento e espelhamento.
Texturas	Confeccionadas em 2D	2D aplicadas em 3D

Fonte: Do autor (2012)

Figura 7 - Cenário digital 2D



Fonte: LAZZARI (2005, p. 32)

Figura 8 - Cenário digital 3D



Fonte: CRYTEK 01

A representação de cenas tridimensionais é, naturalmente, bem mais complexa que a de figuras bidimensionais. Cenas tridimensionais podem usar os mesmos conceitos de símbolos e hierarquias das figuras bidimensionais, o que implica a utilização de transformações lineares tridimensionais. Estas incluem, assim como no caso bidimensional, a translação, a rotação, a mudança de escala, o cisalhamento e o espelhamento (PAULA FILHO, 2000, p.159).

As texturas aplicadas nos polígonos das imagens 3D, para torná-las mais próximas a realidade, nada mais são que quaisquer imagens 2D. É importante salientar que a aplicação de texturas não precisa ser exclusivamente em superfícies planas, pois é perfeitamente viável adaptá-las em superfícies curvas e esféricas, como se pode observar nas Figuras 9 e 10.

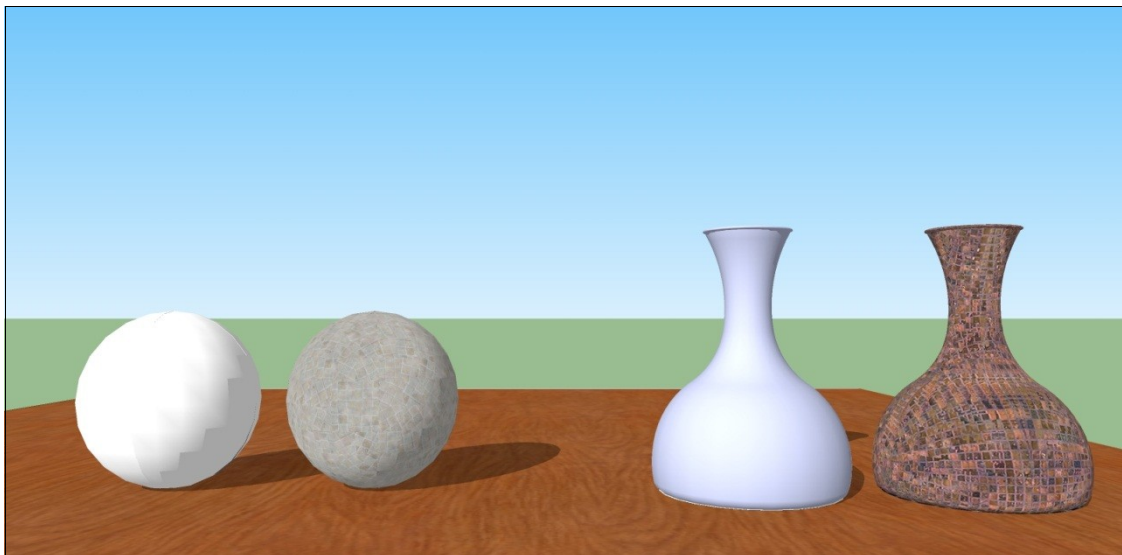
É importante frisar também, que os jogos digitais 3D não são constituídos em sua totalidade por modelos tridimensionais, na sua composição também são necessárias imagens bidimensionais que, geralmente são aplicadas como texturas, para compor a interface gráfica, com janelas, botões, barras de energia, dentre outros componentes gráficos.

Figura 9 – Textura



Fonte: CRYTEK 02

Figura 10 - Textura



Fonte: Do autor (2012)

Vê-se que as imagens 2D e 3D, tem suas diferenças principalmente em relação a percepção visual, mas também que elas podem ser usadas em conjunto e o resultado é um realismo visual cada vez mais próximo a perfeição.

2.3 IMAGENS VETORIAIS E BITMAP

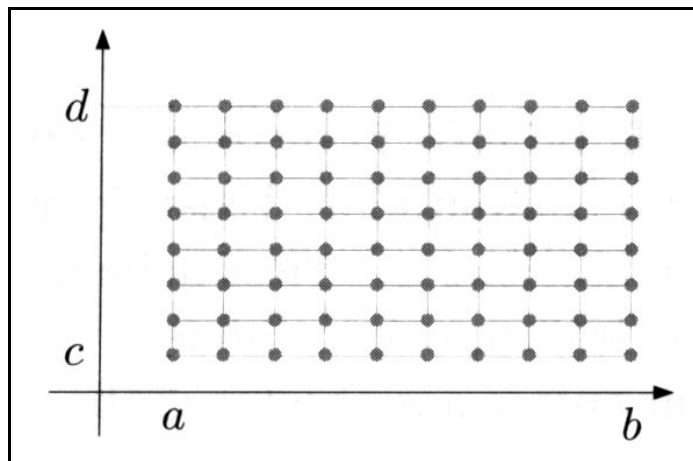
De acordo com os níveis de abstração contínuo e discreto, no Universo de Representação, existem dois formatos básicos para armazenar os objetos gráficos no computador: o formato vetorial e o formato matricial.

O formato vetorial descreve os atributos que definem o objeto no domínio

contínuo. A informação é representada em um espaço topológico e são definidas funções contínuas nesse espaço por um conjunto de segmentos de retas, curvas, vértices de polígonos e malhas de controle de curvas ou superfícies paramétricas descritos pelas coordenadas de posição (pontos iniciais e finais) ou ações no espaço. Este formato é utilizado, em geral, para descrever a estrutura geométrica de seus objetos gráficos sendo que as informações geométricas precisam ser complementadas por informações topológicas para especificar completamente o modelo de um objeto gráfico.

O formato matricial (raster ou bitmap - mapa de *pixels* ou de bits) descreve os atributos que definem o objeto no domínio discreto e é obtido pela discretização uniforme do espaço no qual está definido o objeto gráfico. Neste espaço é definido um reticulado bidimensional, também chamado de matriz de discretização, e cada célula determina uma amostra do objeto gráfico (Figura 11).

Figura 11 - Reticulado uniforme da representação matricial bidimensional no plano

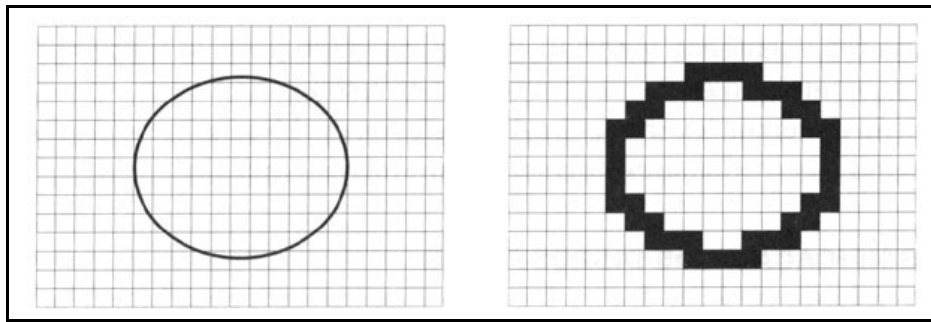


Fonte: Gomes e Velho (1998)

Na informação obtida na representação da matriz, cada elemento é uma estrutura de dados associados à cor e outras componentes da imagem e denominada de pixel.

Na Figura 12, à esquerda, mostra uma elipse representada no formato vetorial, juntamente com um reticulado de discretização de resolução 21 x 15. Na figura à direita, a mesma elipse representada no formato matricial.

Figura 12 - Representação vetorial e matricial de uma elipse



Fonte: Gomes e Velho (1998)

Os processos de conversão entre formatos são denominados de

- Rasterização: formato vetorial para formato matricial
- Reconhecimento de padrões Matricial _ Vetorial

Muitos formatos diferentes são usados para a representação de imagens em arquivos digitais e são diferenciados pelo tipo de elemento de composição (vetor ou mapa de bits), número de cores suportadas, possibilidade de compressão, software que gerou, sistema operacional de suporte, entre outras características. Tratando-se ainda de armazenamento de imagens digitais, dependendo do tipo de imagem a ser armazenada pode-se ter a seguinte classificação, com seus os principais formatos de arquivo associados (PAULA FILHO, 2000):

- Vetoriais (AutoCAD DXF, Microstation DGN, CorelDraw CDR);
- Bitmaps (BMP, PCX, GIF, TIFF, JPEG, PNG);
- Metafiles = Vetorial + Bitmaps (MWM, PNG).

2.4 TÉCNICAS DE COMPUTAÇÃO GRÁFICA E REALISMO VISUAL

Para Azevedo e Conci (2003), o objetivo final da CG é a criação de imagens sintéticas realistas. Sintetizar uma imagem (uma cena ou um objeto) é gerá-la considerando a definição dos dados dos objetos que a compõe. A qualidade de realismo em uma imagem está associada ao seu grau de convencimento, isto é, qual próxima está sua representação da realidade. O realismo visual em CG é composto por técnicas de tratamento computacional aplicada aos objetos gerados (por modelagem de sólidos, partículas, fractais ou outra técnica) com objetivo de criar-lhes uma imagem sintética o mais próxima da realidade com alto grau de verossimilhança como se tivessem sido construídos, fotografados ou filmados. (PAULA FILHO, 2000).

Existe uma variedade de características que proporcionam o realismo em imagens

evidenciando a complexidade do mundo real como pode ser observada na Figura 13 abaixo.

Figura 13 - Complexidade do mundo real: cores, formatos e texturas



Fonte: Correa e Gomes (2004)

Esta complexidade gera dificuldades na representação computacional de objetos e cenas e quanto maior o seu realismo aumenta-se também seu tempo de processamento e custo de geração. De forma simplificada, as principais características de realismo podem ser agrupadas em elementos distintos: a superfície representada por uma malha, isto é, uma área subdividida em vários elementos de área (subpolígonos) definindo a modelagem geométrica, representação, posição e orientação do objeto responsável pela sua **forma**; a **cor** que caracteriza e comunica cada um dos elementos de área; as fontes de **iluminação** que determinam a luz, reflexão, transparência e sombreamento dos objetos; e a **textura** visa descrever as propriedades da superfície de um objeto como características de rugosidade e alguns padrões de repetição mais complexos não cobertos pela geometria ou pelos modelos de iluminação.

O realismo visual, como sinônimo de boa qualidade, é fundamental e perceptível em aplicações como simulação, cenários e personagens de filmes e jogos, realidade virtual, criação de ambientes virtuais e reconstruções realísticas tridimensionais. Podem ser consideradas duas etapas para a aplicação do realismo em imagens: a estática e a dinâmica. A etapa dinâmica visa o movimento de cenas e personagens contemplados na animação computacional. Na etapa estática, foco desta pesquisa, são estudadas e descritas as técnicas e teorias responsáveis por gerar objetos e cenas estáticas com realismo fotográfico ou fotorrealismo.

O processo de criação sintética de imagens é denominado *rendering* que consiste

em converter dados em uma imagem realística ou criar um objeto ou cena a partir de um modelo associando características de realismo fotográfico. O *rendering* é um processo incremental que sintetiza uma imagem a partir da geometria da cena, das informações sobre os materiais de que são feitos os objetos (suas cores e suas texturas), das condições de iluminação ambiente e da posição de observação da cena (AZEVEDO E CONCI, 2003). Essa forma de criação de cenas realísticas é denominada de realismo por passadas e permite que os atributos de uma cena sejam renderizados separadamente da sua geração ou modelagem e, em muitos casos que, diversas técnicas e softwares participem do processo de inclusão do grau de realismo desejado a cena.

A renderização por passadas permite (GOMES e VELHO, 1998): economia de memória (nem todos os objetos são colocados de uma vez para o render); facilidade da introdução de modificações (pode-se alterar somente alguns elementos de uma cena sem perder ou refazer todo o processo de renderização anterior); maior utilização das imagens estáticas; e mixagem com objetos ou texturas reais obtidas por captura possibilitando integração com imagens fotográficas, acrescentando sombras, texturas complexas ou elementos do mundo real (como rostos) à cena.

O processo de renderização com foco no realismo visual envolve sete fases distintas:

- 1) construção do modelo;
- 2) transformações lineares;
- 3) eliminação de polígonos ou faces escondidas (culling back faces);
- 4) recortes (clipping);
- 5) rasterização;
- 6) tratamento de partes escondidas (hidden);
- 7) coloração, textura e iluminação.

Nem sempre são usadas as sete fases em todas as aplicações. Busca-se aplica-las até que se atinja um nível de realismo adequado ao seu uso. Principalmente porque ao aumentar-se o realismo de uma cena aumenta-se também seu tempo de processamento e custo de geração.

2.4.1 Realismo e o Modelo

A Modelagem Geométrica, também chamada de geometria computacional, atua no campo da modelagem de sólidos, atrelando geometria com computação. Para realizar a modelagem geométrica são utilizados métodos que visam descrever a forma e as características geométricas de um objeto. Ela provê uma descrição ou modelo muito mais analítico, matemático e abstrato que o real. Cria-se um modelo do objeto ou processo real sendo este modelo além de mais conveniente e econômico, mais fácil e prático de realizar análises e testes. Esta modelagem serve de base em sistemas *Computer Aided Design (CAD)/ Computer Aided Manufacturing (CAM)*, computadores gráficos, arte por computador, visão por computador, simulação, robótica e animação. Avanços em quaisquer destes campos dependem de quão bem definidos estão os modelos geométricos criados (WEINFURTER, 2003).

Métodos de modelagem geométrica são uma síntese de técnicas de muitas áreas como cálculo vetorial, geometria analítica e descritiva, teoria de conjuntos, análise numérica e matemática matricial, entre outras. Esta combinação de ferramentas matemáticas com a complexidade dos modelos geométricos requer grande esforço computacional (CASTRO FILHO, 2007).

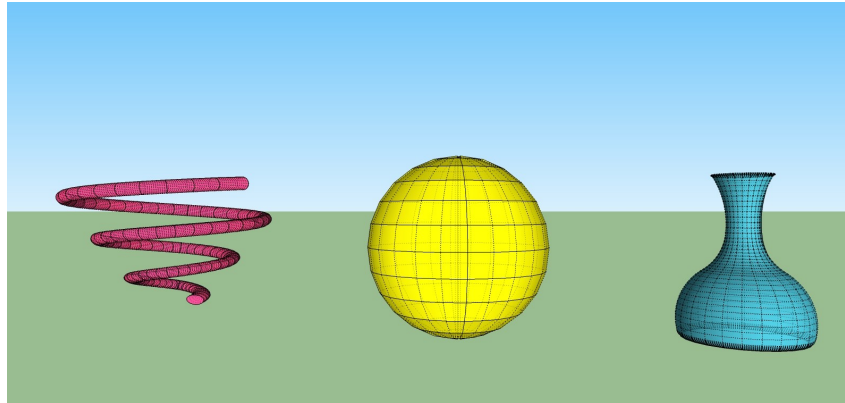
A primeira etapa da renderização com foco no realismo visual é a construção do modelo que conterá todas as informações para o processo de realismo visual, para tanto são utilizadas técnicas de modelagem, para se realizar uma imagem ou um projeto de animação tridimensional. Nesta etapa, é imprescindível um conhecimento aprofundado das ferramentas de que o software dispõe, mas além do conhecimento, será necessário ter disciplina e atenção, para demonstrar ao máximo a criatividade do modelador, também são definidos os principais atributos dos objetos, como sua geometria e expressão.

Cada formato de modelagem possui características que podem facilitar a construção de determinada geometria (BOARDMAN, 1999). Os métodos oferecidos pela maior parte dos softwares de modelagem tridimensional podem ser divididos em cinco categorias (LUCENA, 2002): formas primitivas, modelagem de forma livre, geometria sólida construtiva, modelagem por procedimento e por derivação. Seguem as categorias:

- a) Formas primitivas: constitui uma coleção de formas pré-construídas, que possuem diversos parâmetros de configuração formal. As formas mais comuns são: esferas, cubo, cilindro e cone, como a Figura 14 demonstra abaixo, contudo cada software

pode oferecer sua diversidade de primitivas.

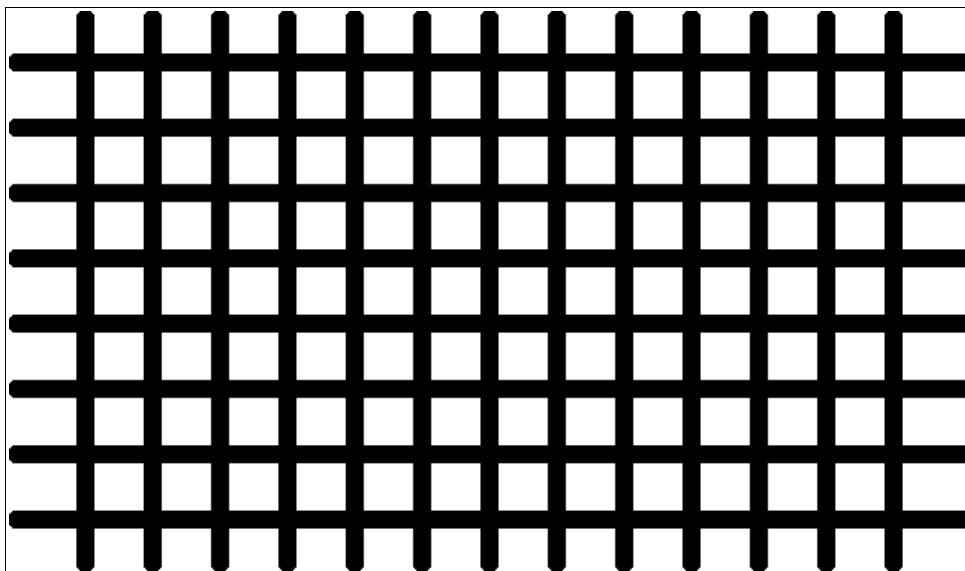
Figura 14 - Formas primitivas



Fonte: Do autor (2012)

- b) Modelagem de forma livre: este método permite que o usuário trabalhar com pontos isolados da malha ou em conjunto desses pontos, abaixo Figura 15, demonstra esse tipo de modelagem:

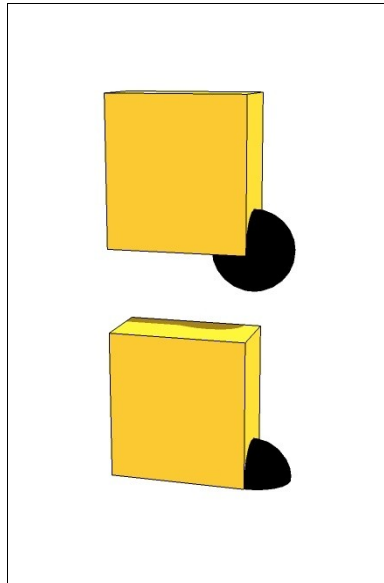
Figura 15 - Modelagem de forma livre



Fonte: Do autor (2012)

- c) Geometria Sólida Construtiva: opera-se a união, intersecção e diferenças entre duas geometrias, para a geração de novas formas 3D, como a demonstrada na Figura 16, chamadas de operações *booleanas*.

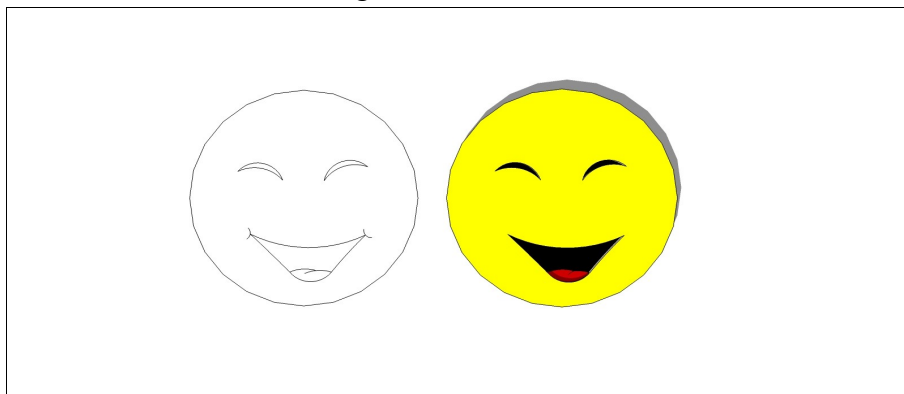
Figura 16 - Geometria Sólida Construtiva



Fonte: Do autor (2012)

- d) Modelagem por procedimento: usa-se um determinado algoritmo, para atribuir um tipo de procedimento, ou ação que define como o objeto será gerado. Nesse método, há diversos parâmetros a serem configurados, o que facilita alterações no formato do objeto.
- e) Modelagem por derivação: esse método pode ser subdividido de três formas para geração de modelos tridimensionais, contudo todas realizam relações entre duas ou mais formas 2D para criação de um objeto 3D:
- extrusão: permite que uma cópia da forma bidimensional seja estendida ao longo de um eixo espacial selecionado, como representado na Figura 17 abaixo:

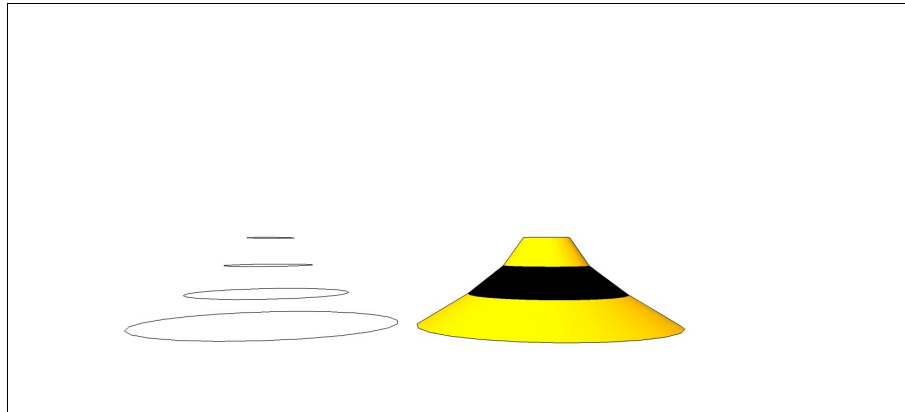
Figura 17 - Extrusão



Fonte: Do autor (2012)

- seção transversal serial: admite a conexão de diversas seções 2D que podem ter diversos formatos e dimensões, como ilustra a Figura 18, abaixo, sendo útil para criação de formas orgânicas.

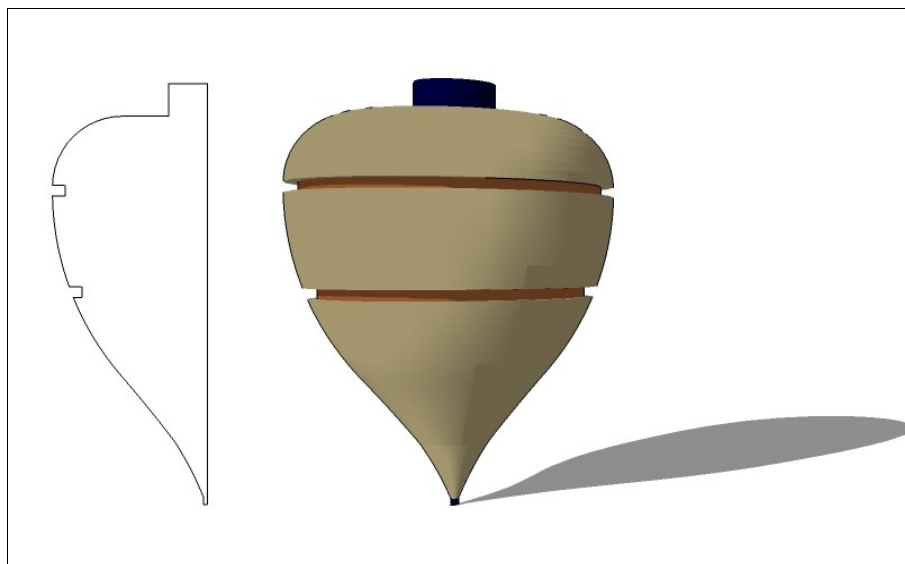
Figura 18 - Seção transversal



Fonte: Do autor (2012)

- revolução: um software executa um giro completo em torno do próprio eixo de uma forma de duas dimensões, como representado na Figura 19 , abaixo:

Figura 19 – Revolução



Fonte: Do autor (2012)

Na segunda etapa da renderização com foco no realismo visual são aplicadas transformações lineares no modelo de modo que ele tenha aparência tridimensional nos diversos dispositivos (geralmente bidimensionais). Essa fase consiste, então, na utilização de técnicas para projeção na vista em perspectiva. Nesta fase deve ser utilizado o conceito de

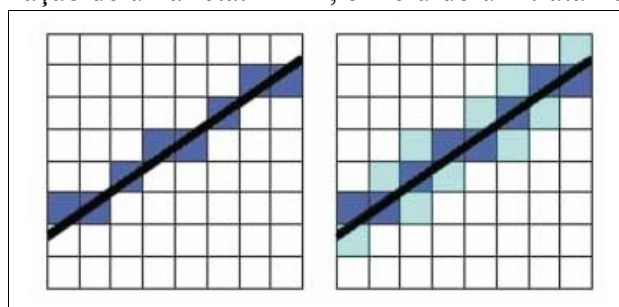
ponto de fuga. Dependendo do ponto de vista que se quer simular, poderão ser utilizados de um a três pontos de fuga. Somente após esta etapa, será possível fazer a eliminação das faces ou polígonos escondidos, ou seja, que não ficariam visíveis para o observador se este fosse um objeto real, ou seja, não virtual (PAULA FILHO, 2000).

A terceira fase da renderização com foco no realismo visual considera a eliminação de linhas, polígonos, superfícies ou faces escondidas devido a posição relativa entre os objetos da cena e o observador (*culling back-faces*). Usa-se o *culling* para a determinação das partes visíveis dos objetos, visto que não há a necessidade de reconstituição do desenho nas partes invisíveis. Uma boa solução para a questão de visibilidade é o principal passo do processo de criação de cenas realísticas. Alguns algoritmos são usados para determinar as linhas, arestas, superfícies ou volumes que são visíveis ou não para um observador localizado em um ponto específico no espaço e, atualmente, são implementados nas placas gráficas.

Na quarta fase da renderização com foco no realismo visual são realizados “recortes”, ou seja, são desconsideradas as partes das cenas que não serão mostradas, chamado de *clipping*. O *clipping*, define os objetos que estão na área de visão do usuário de maneira completa, caso apenas uma parte de algum deles esteja visível somente esta será redesenhada e os que não aparecerem não serão remontados, é usada também para evitar que elementos que estejam afastados da visualização do usuário precisem ser redesenhados, por exemplo, o que está por trás de objetos, detalhes muito longe para serem bem percebidos ou coisas muito pequenas que não merecem ou conseguem chamar a sua atenção.

A quinta fase da renderização com foco no realismo visual realiza a conversão da representação tridimensional para pixels, processo é chamado rasterização, ilustrada na Figura 20. A conversão de coordenadas leva os dados do modelo para o mundo digital onde linhas e áreas serão transformadas em conjuntos de pixels.

Figura 20 - Rasterização de uma reta. Em B, é incluído um tratamento de anti-aliasing



A

B

Fonte: Azevedo e Conci (2003)

Rasterização de retas: geralmente os gráficos são definidos através de primitivas geométricas, tais como: pontos, retas, círculos, textos etc. Ao tentarmos desenhar uma reta no vídeo, deve-se levar em consideração, nem sempre terá uma forma reta perfeita, pois ela será desenhada pelos pixels que puderem ser acessados no dispositivo de visualização utilizado, através de uma aproximação a ser obtida com a utilização do quadriculado formado pelos pixels. A aparência da reta pode ser serrilhada dependendo da inclinação traçada, essa aparência é chamada de aliasing. Ela ocorre em função das quebras de continuidade impostas pela malha de pontos. Esses serrilhados podem ser melhorados através da aplicação de algoritmos de anti-aliasing, que tem o objetivo de tentar “enganar” a visão do observador, fazendo com que as bordas de um desenho fiquem menos nítidas (AZEVEDO E CONCI, 2003).

A sexta fase trata-se da eliminação de partes de um objeto que devem ser removidas, devida à interferência dos diversos objetos presentes na cena, onde, pode ocorrer que uns fiquem na frente de partes de outros, conforme suas posições. Essa fase de renderização com foco no realismo visual é, de certa maneira, uma continuação da terceira, na qual também são eliminadas partes de um objeto que devem ser removidas. Alguns autores englobam estas duas partes no que se chama tratamento de partes escondidas (*hidden*).

Na sétima fase se colore cada pixel individualmente, usando um esquema incremental ou interpolador de sombreado. Deve-considerar as luzes presentes na cena, suas intensidades e direções, assim como todas as características das superfícies representadas: transparência, brilho, reflexão e textura. Mas também as sombras que dos vários componentes fazem entre si e nas superfícies em que se apoiam. O de realismo dessa fase pode ser tão sofisticado quanto à aplicação precisar, sendo necessário o uso de modelos físicos (de representação da luz e sua interação com os materiais) para um tratamento adequado. (AZEVEDO E CONCI, 2003).

2.4.2 Realismo e a Aparência

Com o modelo definido, a próxima fase da renderização com foco no realismo visual busca tratar e perceber e o realismo “fotográfico”. Nesse sentido, são aplicadas técnicas que possibilitem levar em conta fontes emissoras de luzes presentes na cena e suas intensidades e direções (em relação aos objetos). São também consideradas as características das superfícies representadas: cor, transparência, brilho, reflexão e textura. A observação de sombra que os diversos objetos fazem entre si e nas superfícies em que se apoiam movimento

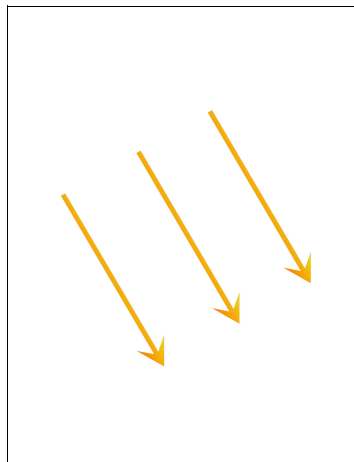
e demais quesitos, para a obtenção de um trabalho que concluído apresente-se o mais fiel com a realidade. O nível de realismo dessa fase pode ser tão sofisticado quanto à aplicação precisar.

O uso correto da cor apresenta vários aspectos importantes na geração de imagens com realismo (GOMES e VELHO, 1998): melhora a legibilidade da informação, possibilita rápida e correta assimilação da informação, permite indicar mecanismos de segurança, permite focar a atenção do observador, permite passar emoções, torna o processo de comunicação mais eficiente. Os três tipos principais de colorização são: Constante (*flat*), *Gouraud* e *Phong*, citados em ordem crescente de aperfeiçoamento do realismo visual. Além disso, há recursos de iluminação, como a reflexão difusa, em que não a fonte de luz não é aparente e a reflexão, em que objetos mostrados com a distorção característica de como são reproduzidos em superfícies espelhadas como vidro ou metal (PAULA FILHO, 2000).

A iluminação que uma imagem na cena pode ocasionar infinitas representações de um objeto pela variação na direção ou intensidade de luz que ele está recebendo. No realismo efeitos que tornam as imagens reais, como reflexões, transparências, texturas e sombras são tratados como modelos de iluminação. As luzes são responsáveis por interagir com o material dos objetos, logo as tonalidades de cor do objeto variam conforme a posição da luz emitida sobre o mesmo. Dentre os diversos modelos de iluminação existentes, destacamos o de Phong no qual são considerados três componentes de iluminação:

- *Directional Light*: como pode-se observar na Figura 21 são raios de luz emitidos para uma determinada posição, estes são infinitos e paralelos.

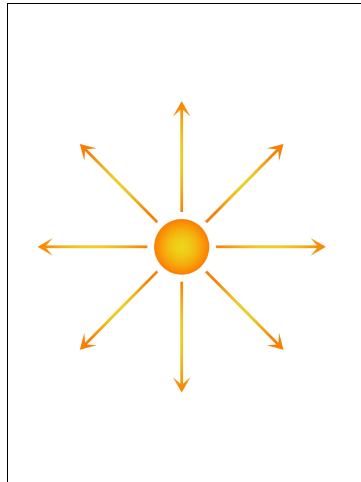
Figura 21 - *Directional light*



Fonte: Do autor (2012)

- *Point Light*: fonte emissora de luz disposta em um determinado local do ambiente, que conforme é mostrado na Figura 22, emite raios de luz para todas as direções.

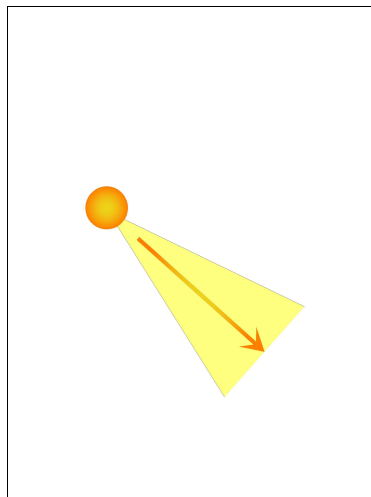
Figura 22 - *Point light*



Fonte: Do autor (2012)

- *Spot Light*: nesse tipo de luz a posição dela no ambiente emite raios de luz em forma de cone para uma terminada área como mostrado na Figura 23:

Figura 23 - *Spot light*



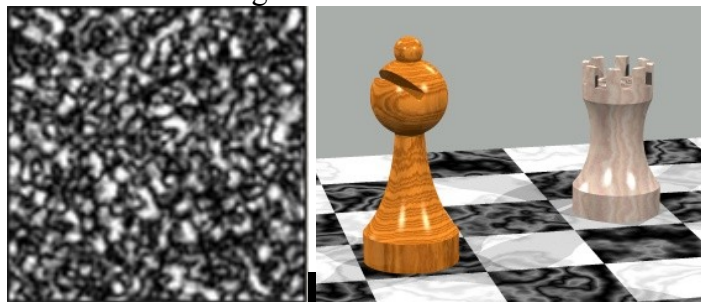
Fonte: Do autor (2012)

Na síntese de imagens baseada somente em modelos de iluminação apenas variações de intensidades das cores são notadas. Estes modelos não são capazes de descrever todas as propriedades da superfície de um objeto, tais como suas características de rugosidade e alguns padrões mais complexos do objeto. A textura é uma técnica que busca dar às superfícies dos objetos características que os façam parecerem mais reais, quando comparados

a simples técnicas de iluminação e sombreamento (por exemplo, a característica de rugosidade). Para reduzir a complexidade de processamento, o acréscimo de detalhes na geometria da superfície é modelado com o uso de mapas de textura a partir de uma função ou mapeamento bidimensional. Dentre as técnicas para mapeamento de textura tem-se, as explicadas e exemplificadas abaixo: mapas procedurais, UVW Map, Texture Map, Mapa de Reflexão, Bump Map, Mip-Mapping, light maps (AZEVEDO E CONCI, 2003). Abaixo estão descritas as técnicas:

- Mapas procedurais: esses mapas dispensam a utilização de imagens e, podem ser inclusive tridimensionais por isso, são muito usados para a síntese de cenas complexas ou aplicações em *real-time* e são boas fontes para mapeamento de textura sintéticas. Dentre os tipos de mapas procedurais, o de ruído, que usa duas cores, mapas sub-procedurais ou uma combinação de ambos para criar um padrão aleatório, conforme Figura 24, tem recebido uma atenção especial, pois têm produzido uma variedade de imagens bastante realistas com texturas geradas por algoritmos fractais.

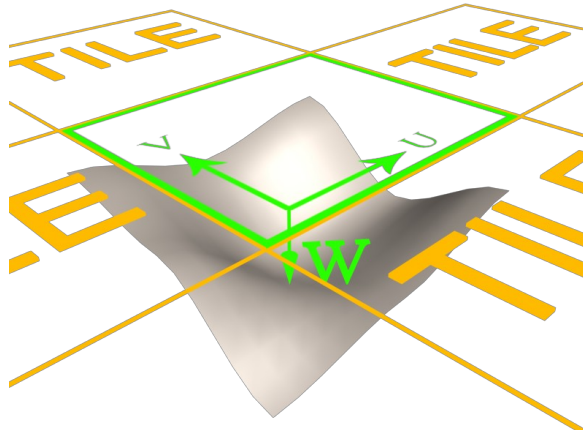
Figura 24 - Texturas



Fonte: Autodesk WikiHelp (2012); Valcar.net (2010)

- *UVW Map*: o sistema de coordenadas usado para descrever o emprego e transformação de mapas é diferente do sistema de coordenada X, Y, e Z usado para descrever o objeto, eles usam as letras U, V, e W (Figura 25); a três letras que precedem as X, Y, e Z no alfabeto empregadas no sistema de coordenadas do objeto. Sendo a letra U o equivalente de X, e representa a direção horizontal do mapa; V o equivalente de Y, representando a direção vertical, W o equivalente de Z e representa a direção perpendicular do plano UV. O eixo W é geralmente usado para mapas procedurais tridimensionais ou para girar os mapas na superfície dos objetos, o segundo caso ocorre quando trocamos os sistemas de coordenadas de VW para WU.

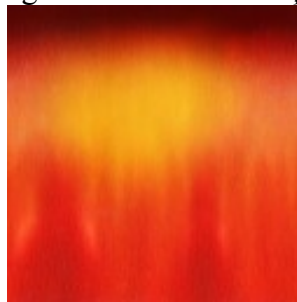
Figura 25 - UVW



Fonte: Texas Tech College of Architecture (2010)

- *Texture Map*: o mapa de textura, como o ilustrado na Figura 26, aplica uma textura sobre um objeto na cena. Pela sua simplicidade e utilidade, passou a ser considerado como padrão para as interfaces de softwares e hardwares gráficos. Os mapas de texturas podem ser usados em cenas complexas com um baixo custo de processamento. Para aplicar a imagem nos objetos, devemos antes seguir alguns passos: a textura deve ser armazenada como um *array*; deve ser mapeada para adaptar-se a superfícies do objeto e a uma visão da cena em perspectiva; ser tratada para retirarem-se imperfeições devidas ao *aliasing*. As técnicas de mapeamento podem também representar objetos transparentes ou semitransparentes. Essa técnica é muito usada na simulação de nuvens e árvores onde mapas 2D são inseridos na cena em retângulos.

Figura 26 - Textura maçã



Fonte: BLEND IT! (2007)

- Mapa de reflexão: é o tipo de mapeamento que reflete na superfície dos objetos os elementos que compõem a cena. Essa técnica pode ser alcançada de duas maneiras: uma delas, tomando um cubo como exemplo, requer seis imagens de textura, uma para cada direção contendo as informações dos objetos que compõem o ambiente, essa técnica não é

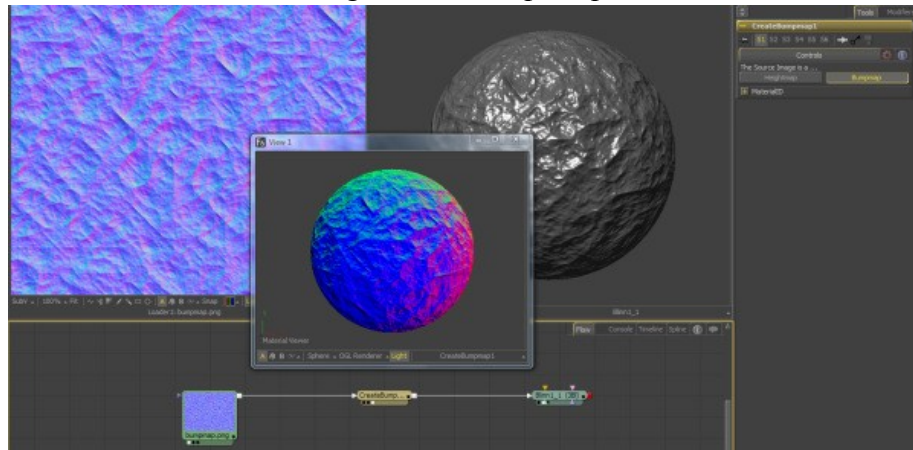
muito exata, porém convincente, o outro procedimento gera uma única imagem de uma esfera refletindo o ambiente. Cada hemisfério da esfera representará um hemisfério do ambiente, um dos problemas dessa técnica é que o objeto não reflete a si mesmo, o que pode resultar em erros na imagem de objetos não convexos. Estes mapas são usados para fazer o render de objetos reflexivos, como a Figura 27 demonstra abaixo, ele é simular efeitos de *ray tracing* a baixo custo.

Figura 27 - Mapa de reflexão



Fonte: Azevedo e Conci (2003)

• *Bump Map*: é uma técnica empregada para acrescentar realismo sem alteração da geometria do objeto, ela adiciona um sombreamento nos pixels, produzindo uma ilusão de relevo no objeto renderizado. A cor de uma superfície está relacionada com ângulo entre o vetor normal da superfície e a direção da luz. No *bump map*, as propriedades de refração da luz são usadas para indicar quais partes são mais escuras ou mais claras. Para isso, a técnica irá perturbar o vetor normal em vários pontos da superfície criando uma ilusão de que algumas partes da superfície estariam elevadas ou rebaixadas, a Figura 28 demonstra o resultado do emprego dessa técnica. Com a finalidade de uma perturbação adequada da normal, para produzir determinado efeito usamos um mapa de altura (*Height Map*), fazendo com que zonas de maior intensidade de cor (branco) pareçam em alto-relevo e as de menor intensidade (preto) pareçam em baixo-relevo, ou vice-versa.

Figura 28 - *Bump Map*

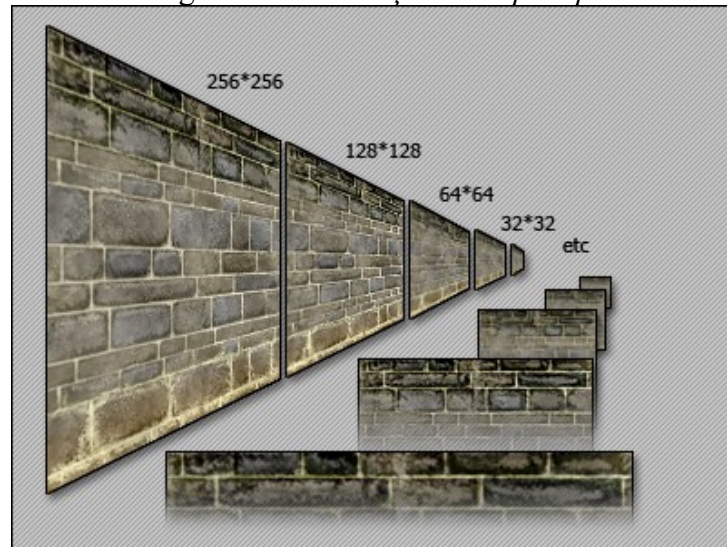
Fonte: HDNLE.com (2010)

• *Mip-Mapping*: é um tipo de mapeamento que indica a solução de dois problemas no mapeamento de objetos em cenas de animação ou *real-time rendering*. O primeiro problema acontece com o afastamento do observante em relação aos objetos. Quando as texturas destes ficam maiores do que a quantidade de pixels disponíveis, o render terá de aplicar o *anti-aliasing*, pois é produzido ruídos de fundo. Além desse indesejado ruído, exemplificado na Figura 29, devemos somar o tempo de processamento para o cálculo da média. O segundo problema ocorre com a aproximação do objeto alvo, revelando detalhes indesejados da textura (geralmente por baixa resolução). Pode se contornar esse problema aumentando-se a resolução da imagem, porém essa solução acarretaria na necessidade de mais memória, tempo de processamento e ruídos provocados pelo primeiro problema. Para solucionar ambos os problemas, o *mip-mapping* indica o uso de mais de uma imagem como textura, onde cada imagem possui um nível de resolução decrescente conforme o observador se afasta, conforme demonstra a Figura 30. Isso significa uma qualidade melhor do render, economizando memória e processador.

Figura 29 - Ruídos



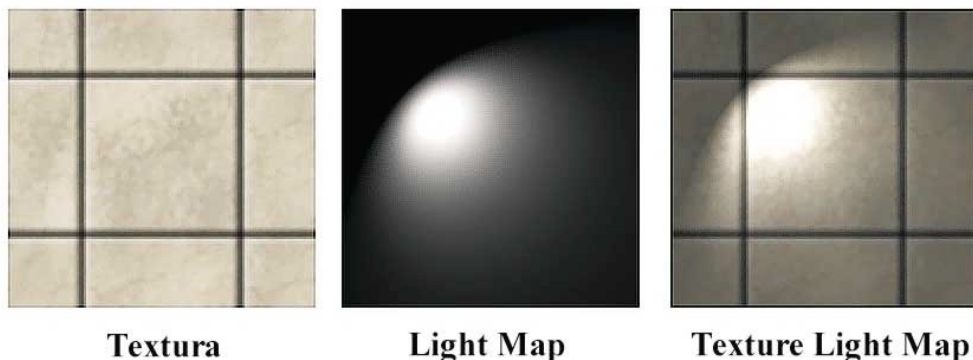
Fonte: Azevedo e Conci (2003)

Figura 30 - Utilização do *mip map*

Fonte: Projectall (2011)

• *Light Maps*: método de armazenamento da iluminação difusa que possibilita a prática de um modelo de iluminação global que pode ser executado em um tempo razoável. Essa técnica é aproveitada recentemente para iluminação e sombreamento em aplicações de tempo real. Os *light maps* possibilitam um pré-cálculo da luz para armazenamento em mapa de textura bidimensional e são usados para dar velocidade no render de objetos 3D com iluminação complexa. Um dos grandes benefícios é a possibilidade de se usar, o resultado de métodos, como radiosidade ou outros de iluminação global, que poderiam demorar até 1 hora para serem calculados. É importante perceber que, quando o observador (ou o objeto) muda sua posição, as coordenadas do mapeamento devem ser corrigidas. Outra importante característica dessa técnica é a capacidade de prover um *highlightshading* mais realístico, sem sacrificar a velocidade de render. Esse benefício é ainda mais acentuado quando lidamos com superfícies reflexivas ou muito brilhantes.

Figura 31 - Fonte de luz aplicada na textura



Textura

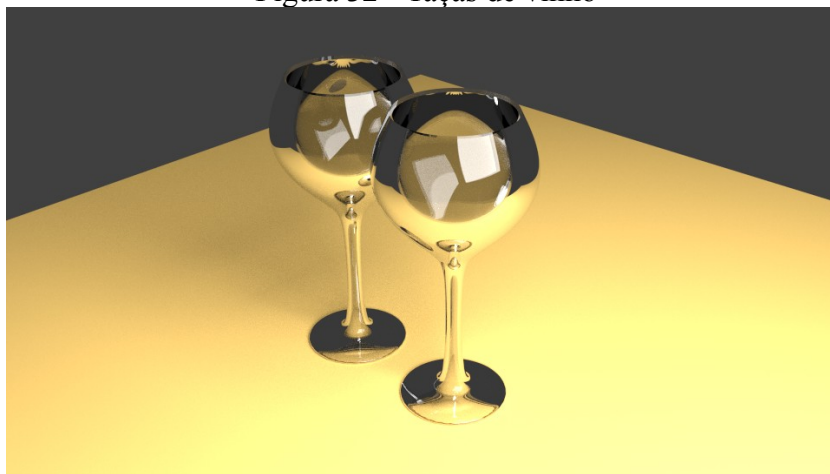
Light Map

Texture Light Map

Fonte: Azevedo e Conci (2003)

Interagindo com a luz, o tipo de material e sua textura, pode-se verificar que certos objetos são transparentes, como ilustrado na Figura 32, ou seja, deixam a luz passar por ele. A luz por propriedades físicas sofre mudanças na sua direção (refrações e reflexões). A aplicação de técnicas de transparência, sombra, reflexão e radiossidade dão um toque de realismo à figura mostrando a influência da luz entre os objetos da imagem vista pelo observador. Para tratar a interação da luz com objetos de forma um tanto realista pode-se utilizar a técnica de *Ray-tracing*.

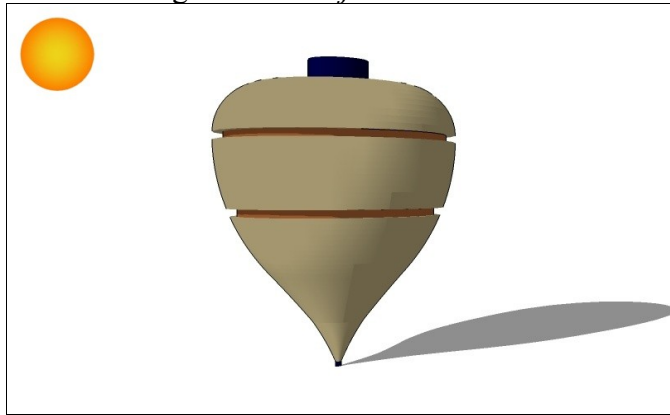
Figura 32 - Taças de vinho



Fonte: Do autor (2012)

A técnica de *Ray-tracing* realiza um cálculo dos trajetos, no ambiente tridimensional, dos raios emitidos por determinada fonte de luz, até o ponto de vista da câmera, com este método torna-se possível visualizar o efeito de fontes de luz. A técnica de *Projected Shadows* projeta a sombra de um elemento sobre a textura de outro, como ilustra a Figura 33 esta sombra é determinada a partir de um ponto emissor de luz do ambiente, que se projeta nos elementos que estejam oclusos ao objeto que emissor da sombra. A técnica de *Radiosity* calcula a transferência de quantidade de luz entre as superfícies de todos os objetos envolvidos na renderização até que a energia de luz seja totalmente absorvida pelas superfícies, ou dissipada pelo espaço. Pode-se perceber que este método calcula a direção dos raios de luz entre todas as direções possíveis para todo o ambiente envolvido, diferente do método *Ray-tracing*, que calcula apenas a direção dos raios entre a câmera e a fonte de luz.

Figura 33 - *Projected shadows*



Fonte: Do autor (2012)

3 JOGOS DIGITAIS

Os Jogos Digitais (JD) são provavelmente a forma de arte mais avançada até agora na história humana, porque eles sintetizam texto, imagem, som, vídeo, e a participação ativa da audiência em uma experiência estética unificada (BROWN, 2008).

Eles abrangem diferentes tipos de recursos, que podem ser observados nos diversos exemplares existentes. Nesse capítulo explanaremos sobre conceitos, tipos, exemplos, mas também sobre a evolução dos jogos computacionais, no que diz respeito ao realismo visual.

3.1 CONTEXTOS E CARACTERÍSTICAS DE JD

Os jogos fazem parte do nosso cotidiano e de modo geral, os utilizamos de várias formas: os dos sentidos, em que a curiosidade nos leva ao conhecimento; os jogos corporais expressos na dança, nas cerimônias e rituais de certos povos; o jogo das cores, da forma e dos sons, presente na arte dos imortais; o jogo do olhar. Enfim, ele está presente em diversos lugares. A intensidade do poder do jogo é tão grande que nenhuma ciência conseguiu explicar a fascinação que ele exerce sobre as pessoas (WINCKLER, 2003).

Objetivando aos Jogos Digitais, entendemos que ele é um produto de comunicação, dado seu fluxo de informação, as suas características narrativas, seu poder de sedução e as diversas formas que ele pode atuar como ação em estratégias comunicativas, sendo ele mesmo a mensagem ou conduzindo uma mensagem da forma mais interativa que um meio digital experimenta (PINHEIRO, 2007).

Os jogos podem possuir temas óbvios nas principais mídias, para muitos os jogos são um mundo misterioso, porque sua cultura é mais viva em espaços culturais menos proeminentes, como sites de resenhas da internet e fóruns. Uma vez descobertos estes espaços culturais, a quantidade de informações específicas do objeto, as opiniões compartilhadas, a linguagem, e inúmeras gírias podem fazer a cultura dos jogos um campo quase impenetrável aos que vêm de fora (TAVINOR, 2009).

Para se desenvolver um jogo existem muitas etapas a serem atravessadas, começando pelos vários desafios de programação, como o conhecimento profundo de tecnologia de ponta sobre bibliotecas gráficas, de sons, entre outras; o domínio de técnicas de programação para abordar diversas dimensões ortogonais de um jogo: por exemplo, linha de tempo, ciclo de eventos, manipulação de imagem e som, interação com o usuário; a

complexidade no desenvolvimento de programas grandes (ALMEIDA, 2004).

Uma das classificações de tipos de jogos são a dos dois grandes grupos 2D e 3D, e o grupo de jogos que utiliza as duas tecnologias. A seguir breve explicação desses grupos.

Os jogos 2D tem o mercado altamente lucrativo, pois possuem um estilo de jogo mais procurado, que são os “casuais”, que caracterizam-se pela simplicidade da jogabilidade e facilidade de aprendizado. O perfil dos jogadores que se interessam por jogos casuais também é um pouco diferente do público que prefere jogos do tipo Triple A¹, pois é formado por adultos com mais de 30 anos, independentes financeiramente e que não passam grandes intervalos de tempo jogando (BATES, 2001).

A maioria dos jogos 2D são conhecidos como *Tile Based Games* porque usam uma técnica muito comum de construção de cenários baseados em mapa de *tiles*. A tradução da palavra “*tile*” é de azulejo. A idéia é que se têm vários blocos de mesmo tamanho, mas com diferentes imagens e que o cenário é montado a partir da junção desses blocos de uma forma organizada. Esta técnica é usada em jogos de plataforma, como o Mario da Nintendo, e até em complexos jogos de estratégia em tempo real como o Warcraft II da Blizzard (ALMEIDA, 2004). Abaixo as Figuras 34 e 35, apresentam jogos em 2D.

Figura 34 - Jogo 2D



Fonte: LION KING

¹ Triple A: Escala máxima de excelência de um produto.

Figura 35 - Jogo 2D



Fonte: TOP GEAR

O jogo 3D é um software mais complexo, visto que é composto por vários elementos como: IA², redes de computadores, módulos de computação gráfica, multimídia³, dentre outros, e mais, satisfazendo uma das características fundamentais de um jogo, que é o funcionamento em tempo real, esses elementos devem operar em total sincronia.

Segundo Rollings, o processo de desenvolvimento de um jogo 3D envolve as seguintes etapas: Efetuar o Design; produção de áudio; produção das imagens em 2D; modelação; seleção do motor de jogo; integração dos aspectos artísticos com os aspectos computacionais (ROLLINGS, 2000).

Os jogos tridimensionais, como o da Figura 36, envolvem computação gráfica e modelagem, esse tipo de jogo pode ser capaz de construir muitos dos elementos necessários à concretização visual dos jogos eletrônicos, possibilitando a sua manipulação, bem como a aplicação dos conceitos artísticos ao mesmo tempo e por meio de um conjunto de ferramentas tecnológicas (COUCHOT, 2003).

² IA: Inteligência artificial.

³ Multimídia: Multimídia consiste na integração, controlada por computador, de textos gráficos, imagens, vídeo, animações, áudio e outras mídias, que possam representar armazenar, transmitir e processar informações de forma digital (MARSHAL, 2001).

Figura 36 – Jogo 3D



Fonte: BATTLEFIELD

Há uma imensa quantidade de tipos de jogos e a cada dia surgem novos sucessos, para todos os gostos e com utilização de diferentes técnicas, estéticas, gêneros, que vão evoluindo e sendo aperfeiçoadas. Esse fato é uma maravilha aos adeptos a esse tipo de entretenimento e um convite atraente aos que ainda não são.

3.2 EVOLUÇÃO DO REALISMO NOS GRÁFICOS

Mas o que seria realismo diante da nossa visão? A definição da realidade captada por nossos olhos foi definida por experts como uma imagem formada por 80 milhões de polígonos (BOLTER, 2000).

Realismo e estilização são puramente respostas às perguntas. Exemplo, se um designer se aproxima de um artista, com um jogo de estratégia de Guerra, baseado na Primeira Guerra Mundial, o Hiper-Realismo seria uma opção a se considerar, já uma solução no estilo da *Disney* seria considerado cômico para o público em potencial, por não combinar com o tema do jogo e não trazer a seriedade que o mesmo pede. Porém, aqui é onde uma falsa dicotomia é revelada. Estilização não é relativa apenas a formas ou volumes, mas pode ser

expressa também em tons e texturas, e no caso de um jogo de Guerra, esta seria uma opção viável (MORRIS, 2007).

O realismo nos gráficos começa a evoluir a partir da década de 1980, ocorre uma mudança na concepção da animação pelo surgimento das técnicas de modelagem em 3D. Tais condições possibilitaram tanto fornecer as informações que descrevem o movimento dos objetos, quanto para usar o computador para sintetizar as informações a partir de algoritmos (COUCHOT, 2003).

Com o surgimento das técnicas de modelagem em 3D, uma nova categoria de profissional surge neste período e ganha destaque junto ao ilustrador, que é o artista 3D. Este profissional recebe as idéias visuais propostas pelo artista de modelos 2D na etapa de arte conceitual e trabalha em modelos virtuais tridimensionais, aproximando a estética visual à realidade perante os olhos do público, possuindo características que se parecem com a de um escultor, mas utilizando-se de ferramentas digitais, ao mesmo tempo que trabalha em parceria com o artista 2D, que agora ganha uma nova função além de criar conceitos visuais: a função de elaborar texturas a serem aplicadas nos modelos 3D, envelopando-os, deixando com características foto-realistas. “Trabalhando em conjunto, estas equipes podem criar sinfonias de design em que todos os elementos – personagens, cenários, itens e armas – relacionam-se e se misturam sem problemas.” (JENISCH, 2008, p. 14).

No final dos anos 80, o campo do realismo digital começou a se desenvolver graças aos avanços gráficos disponíveis pelos computadores da época. O uso de sofisticados efeitos especiais e a iluminação por computador tornaram-se prioritários dentro dos Jogos Digitais quando sua utilização crescente tomava maior significação dentro do cinema e da televisão (CHONG, 2010).

Essa preocupação ocidental com o realismo regressa ao passado dos artistas gregos e romanos que procuravam uma versão idealizada da natureza. Este realismo é o ponto de partida de parte da Fantasia e Ficção Científica ocidental, possivelmente pelo desejo de fazer que algo que não exista se torne tão próximo de parecer crível. Do ponto de vista dos jogos, isto faz sentido perfeito se o desejo é criar uma simulação realista da fantasia. Isto também permite equipes de artistas a trabalharem num estilo que é comumente entendido. “As tradições orientais, no entanto, sempre foram voltadas para a estilização, como podemos ver em diversos títulos para consoles, no estilo mangá, como Zelda (Figuras 37 e 38) e Pokémon.” (MORRIS, 2007, p. 64).

Figura 37 - Zelda 2D



Fonte: ZELDA

Figura 38 – Zelda 3D



Fonte: ZELDA

A qualidade do fotorrealismo dentro dos Jogos Digitais se desenvolve pela captação por parte do artista 3D de idéias de geração de imagens com base na fotografia, na pintura hiper-realista e na escultura. Combinando elementos destes campos das artes, estas imagens trazem uma nova luz à consideração da forma, com uma complexidade específica de produção e convencimento de que aquilo que é visto denota o real (LIESER, 2010).

No ano de 2001, a observação e o registro de movimento já estavam completamente digitalizados, onde a função de seguir os movimentos e registrá-los como dados passa a ser do computador. Com isso, reduziu-se a distância entre os movimentos de atores reais e suas representações em 3 dimensões nos Jogos Digitais (CHONG, 2010).

Mas grande parte do crescimento artístico e conseqüente evolução do realismo se deve a tecnologia: o presente é uma época de consoles de última geração e computadores pessoais poderosos - dispositivos que são capazes de criar sofisticados, sensíveis, e cada vez mais lindos mundos ficcionais em que os jogadores penetram para jogar os jogos. O lote mais recente de consoles - *X Box 360*, *Nintendo Wii* e *Playstation 3* – são, cada um, maravilhas tecnológicas que trazem animação digital em tempo real para a casa dos usuários, onde, há menos de 20 anos atrás, animação desta qualidade era de domínio exclusivo dos produtores dos filmes de grandes orçamentos (TAVINOR, 2009, p. 6).

Outro agente que impulsiona o crescimento do realismo são os novos e os maduros públicos dos jogos que estão exigindo mais variedade e também estão prestando uma atenção cada vez mais crítica aos jogos, o que explica um pouco da recente evolução artística (TAVINOR, 2009, p. 7).

Essa influência tecnológica é visivelmente marcante no gênero *First-Person Shooter*, que se define como simuladores de tiro, vistos em primeira pessoa, que geralmente se utiliza da busca de um conceito de fotorrealismo. Este gênero trabalha com a premissa da visualidade em 3 dimensões, apresentada nos Jogos Digitais no início dos anos 90, dado o ponto de vista do jogador dentro do jogo e uma suposta euforia por simulações de realidade virtual que marcaram a época (HUTCHINSON, 2008).

Iniciou-se em *Wolfenstein 3D* (Figura 39), em 1992, jogo que, com uma base de cálculos da engine disponível, criava uma ilusão de ambiente 3D ao impor uma dimensão de profundidade e perspectiva aplicada a gráficos originalmente criados em 2D, causando furor nos jogadores da época, mesmo com personagens e cenários em baixa resolução (MÜLLER, 2011).

Figura 39 - Wolfenstein 3D



Fonte: WOLFENSTEIN 3D

Em 1993 o jogo Doom, ilustrado na Figura 40, trouxe inovações: a engine já passou a trabalhar com a composição de cenários tridimensionais, onde o jogador podia transitar seguindo as dimensões de largura, profundidade e altura, com gráficos de baixa resolução, mas, embora o jogador pudesse subir e descer dentro do cenário, os personagens do jogo ainda eram concebidos por *sprites*⁴ em 2D, ou seja, o jogo era parcialmente 3D (MÜLLER,2011).

Figura 40 – Doom



Fonte: DOOM

⁴ Sprites: são os personagens que aparecem no jogo, que interagem uns com os outros e com o cenário. (ALMEIDA, 2004).

Quake, em 1996 inaugura então uma adequada adaptação mais completa da vivência de ambientes 3D ao gênero, com personagens produzidos em 3 dimensões (MÜLLER,2011). A série Silent Hill, ilustrado na Figura 41, iniciada em 1999 é um exemplo deste gênero de narrativa, que explora a capacidade gráfica ao definir detalhes de criaturas diabólicas como um elemento de suporte para a compreensão da trama (MÜLLER,2011).

Figura 41 – Silent Hill



Fonte: SILENT HILL

Os jogos Medal of Honor (1999), apresentado na Figura 42, e Far Cry (2004), cujas franquias são referência quando se discutem questões gráficas ligadas ao foto-realismo. Se comparados, Wolfenstein 3D, Doom, Quake e Far Cry são um exemplo da evolução da qualidade artística acompanhada dos recursos tecnológicos, servindo de apoio à definição da identidade de um gênero de Jogo Digital (MÜLLER,2011).

Figura 42 – Medal of Honor



Fonte: MEDAL OF HONOR

O realismo também passa a ser relevante quando a narrativa considera o foco nas emoções dos personagens, expressas em seus rostos. Em Heavenly Sword (2007), as animações possuem em sua narrativa um forte apelo no que se refere às expressões nos rostos dos personagens, como pode se observar na Figura 43, utilizando mecanismos de captação de movimento aplicados ao rosto de atores, que encenaram dentro de um estúdio para a composição das cenas, com suas expressões captadas nos computadores e transferidas para os rostos digitais dos personagens (COMPUTER ARTS BRASIL, 2007).

Figura 43 – Heavenly Sword



Fonte: HEAVENLY

A seguir, Figuras 44 e 45 demonstram a evolução do realismo visual em jogos bidimensionais de 2009 à 2011 (TORRES, 2010).

Figura 44 – Tatsunoko vs. Capcom: Ultimate All-Stars – 2009



Fonte: TATSUNOKO

Figura 45 - Marvel vs Capcom 3: Fate of Two Worlds – 2011



Fonte: MARVEL

A tecnologia de realismo e alta definição tornou-se um pré-requisito para a maioria dos jogos contemporâneos e apesar de outras obras, como os filmes populares, sentirem a influência dos recentes desenvolvimentos tecnológicos, nenhuma é tão

intimamente ligada à tecnologia digital como os Jogos Digitais. Os jogadores agora são acostumados a jogar em alta definição. Em televisores digitais, monitores de PCs e através de sistemas de som Dolby Home Theater 5.1, e estes, em conjunto com a evolução da plataforma de consoles e jogos de PC, têm tido um impacto significativo na forma como os jogos modernos parecem e soam (TAVINOR, 2009, p. 6).

3.3 GAME ENGINES

Uma *game engine* é o núcleo básico para a criação de um jogo e tem o objetivo de simplificar o processo de codificação, além de fornecer uma série de funcionalidades agregadas. Ela pode ser dividida em diversas sub-engines, como por exemplo, engine gráfica – que será conceituada- engine física, engine sonora, engine de inteligência artificial, etc. (MIGUEL, 2006).

3.3.1 Engines Gráficas

A engine gráfica é um dos principais e mais importantes componentes de uma game engine, pois é através dela que todas as cenas e objeto serão renderizados e uma das principais características de um jogo é a sua qualidade gráfica (MIGUEL, 2006).

As principais engines gráficas utilizadas são o OpenGL (*open source*) e o DirectX. Estas duas APIs⁵ provém uma interface entre o aplicativo e o hardware de processamento gráfico (ZERBST, 2004).

3.3.1.1 OPENGL

O OpenGL (*Open Graphical Library*) pode ser definido como uma API. Pode-se classificá-la como uma biblioteca de rotinas gráficas para modelagem 2D ou 3D, extremamente portátil e rápida possibilitando a criação de gráficos 3D com excelente qualidade visual e rapidez, uma vez que usa algoritmos bem desenvolvidos e otimizados pela *Silicon Graphics* (AZEVEDO, 2003).

A biblioteca OpenGL transcende do desenho de primitivas gráficas, tais como linhas e polígonos, dando suporte a iluminação, sombreamento, mapeamento de texturas e transparência. Além disto, a biblioteca OpenGL executa transformações de translação, escala e rotação, através da multiplicação de matrizes com transformações cumulativas, ou seja umas

⁵ API: (Application Program Interface). É uma interface de software para aceleração da programação de dispositivos gráficos, com aproximadamente 120 comandos para especificações de objetos e operações necessárias para a produção de aplicações gráficas interativas 3D (AZEVEDO, 2003).

sobre as outras (AZEVEDO, 2003).

A OpenGL (*Open Graphics Library*) foi desenvolvida no início da década de 90. Devido a sua performance e recursos disponíveis, sua tecnologia está presente em diversas aplicações, tais como: animações 3D, realidade virtual, jogos, e assim por diante (DALMAU, 2003). É biblioteca gráfica poderosa e de baixo nível que fornece ao programador uma interface para o hardware gráfico. Foi originalmente desenvolvido pela *Silicon Graphics*, empresa responsável pelos efeitos especiais vistos em filmes como *Exterminador do Futuro 2* e *Parque dos Dinossauros* (ALMEIDA, 2004).

Segundo Azevedo (2003), as principais características do OpenGL são:

- a) Aceleração de hardware: Aceleração do processamento geométrico, luzes, transformações e render;
- b) Efeitos 3D em tempo real: *Real-time fog*, *anti-aliasing*, volume *shadows*, *bump mapping*, *motion blur*, transparências, reflexões, texturas 3D, volume *rendering* e outras;
- c) Suporte a inovações futuras de *software* e *hardware*: Um mecanismo de extensão permite a configuração para novos *hardwares* e *softwares*;
- d) Estabilidade: Estações avançadas de supercomputadores vêm utilizando essas bibliotecas desde 1992;
- e) Escalável: Suas aplicações API podem ser executadas de pequenos aparelhos eletrônicos até supercomputadores com a utilização máxima dos recursos disponíveis;
- f) Assim como a OpenGL, o DirectX também é uma API porém, mais completa.

3.3.1.2 DIRECTX

O DirectX é uma API direcionada para o sistema operacional *Microsoft Windows* que foi desenvolvida para prover uma interface para o controle eficiente de hardware multimídia e permitir aos programadores trabalharem com comandos e estruturas de dados de alto nível. Ele é construído sobre uma camada abstrata de hardware, chamada de HAL (*Hardware Abstraction Layer*) que oculta às dependências específicas de determinados dispositivos, o que torna os programas desenvolvidos independentes de dispositivo, ou seja, funcionarão em qualquer hardware, desde que sobre a plataforma Windows (KOVACH,

2000).

Essa API é formada por diversos componentes, cada um representando um diferente aspecto do sistema, que podem ser utilizados independentemente, permitindo ao desenvolvedor utilizar apenas os necessários para a sua aplicação, os componentes são os seguintes (JONES, 2004):

- a) *DirectX Graphics*: Componente que manipula todas as saídas gráficas, provê funções para a manipulação de objeto gráficos 2D e 3D;
- b) *DirectInput*: Toda a interação entre o usuário e o aplicativo é feita através deste componente. Inclui suporte para diversos dispositivos de entrada, como teclado, mouse, *gamepad* e *joysticks*;
- c) *DirectPlay*: Componente que dá suporte a criação de aplicativos para comunicação em rede, permitindo a criação de jogos multiplayer. O *DirectPlay* fornece uma interface de alto nível, não sendo necessário a implementação de todos os aspectos da comunicação;
- d) *DirectSound*: Componente para a manipulação de efeitos sonoros, permite a execução simultânea de um ou mais arquivos no formato WAV, provendo o controle total sobre a mídia em execução;
- e) *DirectMusic*: Permite a criação de trilhas sonoras dinâmicas, baseadas em determinadas regras determinadas pelo desenvolvedor;
- f) *DirectShow*: Provê acesso a execução de vídeos e arquivos de áudio, o principais formatos suportados são: AVI, MPEG, MP3, ASF, entre outros;
- g) *DirectSetup*: Componente para a distribuição dos aplicativos desenvolvidos, permitindo que antes da execução seja verificado se o sistema possui o DirectX instalado e se a versão é compatível, caso não seja a nova versão é instalada.

Concluindo ambas usam conceitos diferentes para cumprir as suas funções de APIs, o OpenGL utiliza o conceito de programação estruturada, já o DirectX é completamente orientado a objeto (ZERBST; DÜVEL, 2004).

3.3.1.3 OUTRAS

Além do OpenGL e do DirectX existem outras engines gráficas que são derivadas

destas duas, com por exemplo, o OGRE3D (*Open-source Graphics Rendering Engine*) que foi escrita em C++ com o objetivo de tornar fácil e intuitivo o desenvolvimento de aplicativos 3D, a biblioteca abstrai muitos dos detalhes complexos da utilização do DirectX e do OpenGL provendo uma interface baseada em objetos e classes de alto nível (LIMA, 2008).

4 TRABALHOS CORRELATOS

Neste capítulo serão apresentados de forma sucinta, outros trabalhos relacionados ao Trabalho de Conclusão de Curso presente, *REALISMO VISUAL: RENDERIZAÇÃO DE IMAGENS EM JOGOS COMPUTACIONAIS*. Abaixo são descritos os trabalhos correlatos:

4.1 ESTUDOS DE TÉCNICAS REALISTAS DE RENDERIZAÇÃO EM TEMPO REAL

Relatório do Trabalho de Conclusão do Curso - Estudo de Técnicas Realistas de Renderização em Tempo Real - apresentado por Rudi Lopes Bravo de Andrade, requisito da disciplina de Projetos I, do curso de Ciências da Computação da Universidade Federal de Santa Catarina, em Florianópolis, no ano de 2007. O objetivo deste trabalho é estudar as técnicas de renderização que foram propostas por diversos autores, analisando o realismo que a técnica propicia e também a sua velocidade de execução. O objetivo deste trabalho foi estudar quais plataformas de *hardware* são viáveis às execuções das técnicas de renderização, especificando quais delas atendem aos requisitos de realismo e tempo real, mas também, propor um meio eficiente de uso de técnicas realistas em aplicações de tempo real, e aplicar a solução no contexto das aplicações do laboratório *Telemedicina*. Ele esclarece alguns aspectos de desenvolvimento de aplicações realistas em tempo real, apresenta um estudo sobre técnicas de renderização que foram propostos por diversos autores, analisando o realismo que a técnica propicia e também a sua velocidade de execução. O método utilizado pelo acadêmico foi fazer um levantamento bibliográfico das técnicas básicas de renderização, de forma generalizada, sem levar em conta os fatores de realismo e tempo real, apenas para familiarização com as diversas possibilidades de complementação, posteriormente focou em métodos que provavelmente atenderiam aos requisitos, procurando os que tinham as características que se mostravam essenciais durante a primeira fase do projeto. Para validação de uma técnica, houve uma etapa de implementação, analisando os resultados obtidos e qualificando-a em aplicável ou não ao problema proposto. Durante essa etapa de validação, foram observadas diversas características dos métodos, e as que não foram satisfatórias na síntese de imagens médicas tridimensionais, existia a possibilidade de classificá-la como mais apropriado à outro contexto de aplicação.

4.2 3D BUILDER: UMA GAME ENGINE PARA A CRIAÇÃO DE JOGOS 3D

Trabalho de Conclusão de Curso apresentado como exigência para obtenção do título de bacharel em Ciência da Computação pela Universidade do Contestado – UNC Núcleo de Porto União, do acadêmico Edirlei Everson Soares de Lima, em Porto União, em 2008, com título 3D Game Builder: Uma Game Engine para a Criação de Jogos 3D. O objetivo deste trabalho foi desenvolver uma *game engine* completa, onde todos os módulos necessários para o desenvolvimento de jogos estivessem reunidos em uma única ferramenta, o que possibilitaria a criação de jogos 3D de maneira rápida e simplificada. A metodologia aplicada no desenvolvimento deste trabalho foi a divisão da *game engine* em diversas sub-engines, como por exemplo, a engine gráfica, engine física, script engine, entre outras. O resultado deste trabalho foi que a *game engine* desenvolvida, chamada de *3D Game Builder*, atendeu a todas as expectativas e requisitos definidos inicialmente, além de oferecer novas funcionalidades que foram agregadas durante o desenvolvimento, fornecendo uma interface simples e intuitiva para pessoas com pouca experiência e oferecendo recursos avançados para desenvolvedores experientes. O *3D Game Builder* foi desenvolvido com o objetivo de ser uma ferramenta completa para criação de jogos, porém não ficou restrito a jogos, podendo ser utilizado para diversas tarefas, como por exemplo, para a simulação de ambientes 3D, *walkthroughs*, entre outras aplicações. Logo, ela se tornou uma ferramenta genérica para criação de qualquer tipo de aplicação 3D. O interessante é que essa ferramenta não se tornou apenas mais uma entre as existentes, se tornou bastante popular entre os desenvolvedores da área, principalmente por ser a primeira do gênero desenvolvida no Brasil. Durante os 6 primeiros meses em que foi disponibilizada para *download* teve mais de 25 mil *downloads*, além de se tornar notícia em diversos sites de desenvolvimento de jogos, informação dada pelo autor, Edirlei Everson Soares de Lima, em 2008.

4.3 PROJETO E DESENVOLVIMENTO DE JOGOS COMPUTACIONAIS

Relatório Final de Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Ciência da Computação, da Universidade do Estado de Santa Catarina – UDESC, em Joinville, no ano de 2006, pelo acadêmico Gean Alex Pereira. Título do trabalho: Projeto e Desenvolvimento de Jogos Computacionais. O objetivo deste trabalho é avaliar os processos envolvidos nas fases de projeto e desenvolvimento de jogos computacionais, percorrendo desde a fase inicial (concepção) até o produto final (jogo). A metodologia de apresentação de trabalho se dá com formas de elaboração e desenvolvimento de jogos

computadorizados, bem como, um estudo de caso utilizando as metodologias estudadas. De acordo com o estudo realizado constatou-se que os procedimentos existentes no projeto são: definição da história, sendo que esta apresenta um relato de como será o jogo, em seguida define-se os mecanismos do jogo que contêm de maneira detalhada todas as interações possíveis, como ações que o jogador poderá executar e que os objetos sofrerão, entre outros.

4.4 OS CONCEITOS ESTÉTICOS-VISUAIS DOS JOGOS DIGITAIS

Dissertação de Mestrado em Comunicação Social para obtenção de título de Mestre, pelo Programa de Pós Graduação em Comunicação Social da Pontifícia Universidade Católica do Rio Grande do Sul, em Porto Alegre, no ano de 2011, apresentado pelo acadêmico Eduardo Fernando Müller. Título: Os Conceitos Estético-Visuais dos Jogos Digitais, o objetivo desse trabalho foi realizar um estudo sobre os conceitos estético-visuais presentes nos Jogos Digitais, adaptando as idéias de estética e pós-modernismo às suas visualidades, compreendendo os fatores responsáveis pela escolha de uma determinada estética visual para um jogo, e definindo categorias para os conceitos estético-visuais existentes. A metodologia adotada neste trabalho permite uma tomada de decisões sobre as particularidades das formas simbólicas a serem abordadas, em três etapas, primeira: análise sócio-histórica para se “reconstruir as condições sociais e históricas de produção, circulação e recepção das formas simbólicas” (THOMPSON, 1995, p. 366). Como uma segunda etapa, a análise formal ou discursiva sugere que as construções simbólicas apresentam uma estrutura articulada, ou seja, são formas baseadas em regras e recursos disponíveis ao produtor, esta etapa da pesquisa é “um tipo de análise que está interessada primariamente com a organização interna das formas simbólicas, com suas características estruturais” (THOMPSON, 1995, p. 369). Última etapa: de interpretação/reinterpretação, procede, em síntese, como um processo de construção de possíveis significados sobre o objeto. Sugere uma interpretação criativa, por parte do autor do trabalho, do que está sendo representado ou do que é dito, sendo mediado pelas duas etapas anteriores. Concluiu-se que um conceito estético-visual relativo aos Jogos Digitais é definido pelos seguintes fatores: a tecnologia, pela sua capacidade de impor restrições técnicas à elaboração das imagens; a narrativa, considerando a história que é contada no jogo, e a escolha de uma determinada estética visual recebe um relevante papel para a sua devida compreensão; o sistema de regras, limitando as possibilidades de interação do jogador, permitindo assim uma capacidade de intervenção estético-visual por parte do mesmo, e considerando que o nível de complexidade das regras interferem na elaboração das

interfaces gráficas; as condições de produção, em que a definição de um conceito estético-visual está adequada às capacidades técnicas da equipe, ao prazo de produção, suas possibilidades de pesquisa e a cultura dos atores de produção; e as adaptações ao mercado e ao público-alvo, considerando que as estéticas visuais podem servir de estratégia de diferenciação dos produtos, direcionando-os de maneira mais adequada a determinados públicos, utilizando-se de diversos recursos de distribuição.

4.5 PROGRAMAÇÃO DE JOGOS 2D USANDO O MRDX E FUNDAMENTOS DE JOGOS 3D

Trabalho de Conclusão de Curso apresentado para obtenção do grau de Bacharel em Ciência da Computação, na Universidade Federal do Pará, no Centro de Ciências Exatas e Naturais, pelo acadêmico Marcos Romero Gonzaga de Almeida, sobre a Programação de Jogos 2d Usando o MRDX e Fundamentos de Jogos 3D, em Belém – PA, no ano de 2004. Neste trabalho apresentam-se técnicas de programação de jogos 2D e 3D. As técnicas 2D são apresentadas com base no MRDX, que é uma biblioteca para programação de jogos proposta por este trabalho, mostra-se também a estrutura de um jogo e os fundamentos envolvidos, como *sprites*, interação com o cenário e controladores, os tópicos de animação e de edição de mapas. O método utilizado neste trabalho, foi primeiramente apresentar todas as funcionalidades do MRDX e como deve ser usado; apresentar a estrutura de um jogo, onde se mostra um simples jogo desenvolvido com o MRDX para que seja explicado os principais elementos que compõe quase todos os jogos; aborda-se sobre tópicos comuns em jogos 2D como sprites, interação com o cenário e controladores; expõe fundamentos necessárias para a programação da animação e mostra uma classe que encapsula estas implementações; são expostas noções de Tile Based Games e um simples editor de mapas é desenvolvido, aborda-se fundamentos de Jogos 3D, visão geral de jogos 3D e comentários sobre OpenGL e movimentação 3D. A versão inicial do MRDX foi desenvolvida com um propósito didático, o ensino de programação de jogos. Como o MRDX implementa os serviços fundamentais, que são triviais para diferentes jogos, um programador inicia diretamente a programação dos aspectos criativos do jogo. Outras vantagens desta biblioteca: ela é livre para download, em português e com código fonte aberto. Com essas funcionalidades do MRDX, neste TCC, foi possível tratar de assuntos de mais alto nível relacionados à programação de jogos.

Estes foram os trabalhos relacionados ao presente TCC, tais referências tem o objetivo de facilitar a avaliação e validar os estudos feitos.

5 REALISMO VISUAL: RENDERIZAÇÃO DE IMAGENS EM JOGOS COMPUTACIONAIS

Para uma avaliação mais precisa das técnicas de computação gráfica relacionadas a texturização em imagens, evidenciando a qualidade do realismo visual e custo computacional, objetivo do trabalho proposto, foram confeccionados modelos gráficos em 2D e 3D, neles foram empregadas técnicas e processos de renderização para a obtenção de características mais próximas do real.

A avaliação das técnicas de texturização na renderização de imagens para jogos computacionais se faz importante, visto que, o Realismo Visual é um quesito importante para o crescimento positivo do mercado da Computação Gráfica.

5.1 METODOLOGIA

Dentro das linhas de pesquisa do curso de Ciência de Computação, esse projeto se insere na linha de Computação Gráfica, pois ela proporciona a criação de ambientes virtuais que com o auxílio de técnicas computacionais cada vez mais avançadas deixam estes ambientes extremamente realistas.

A metodologia aplicada no presente trabalho se dá em duas fases, sendo que a primeira concentrou-se em realizar o levantamento bibliográfico, feita através de artigos, livros e trabalhos acadêmicos. Através desse levantamento e dos conhecimentos obtidos no decorrer do curso de Ciências da Computação, definiu-se o que são imagens de duas e três dimensões assim como a percepção de espacialidade que as mesmas ocupam, fez-se também um comparativo desses dois tipos de figuras, mas também se apresentou os processos, as fases de renderização, assim como as técnicas de CG utilizadas para obtenção do realismo visual, focando na texturização.

Conforme a pesquisa também se apresentou conceitos, tipos, exemplares dos jogos computacionais e citam-se vários títulos de games apreciados por grande público, que também colaboram para demonstrar o quanto o realismo nos jogos eletrônicos evoluíram; também se apresentou as game engines, as engines gráficas. Fez-se exposição resumida de trabalhos correlatos.

A segunda fase foi prática, a execução da mesma se deu a partir dos conhecimentos absorvidos, na fase teórica. Ela foi composta pelos seguintes itens:

- a) Desenvolvimento do protótipo;

- b) Aplicaram-se transformações lineares no modelo;
- c) Identificaram-se, eliminaram-se ou inseriram-se polígonos ou faces ocultas no modelo;
- d) Realizaram-se recorte ou inserção de elementos no modelo;
- e) Converteram-se o modelo em pixels;
- f) Foram tratadas as partes ocultas do modelo;
- g) Foram aplicadas técnicas de coloração, textura e iluminação.;
- h) O protótipo foi renderizado a partir das técnicas aplicadas;
- i) Testou-se o protótipo e foram relatados os resultados;
- j) Para conclusão foi feita avaliação de técnicas de texturização na renderização do protótipo evidenciando a qualidade do realismo visual.

5.1.1 Levantamento Bibliográfico

A primeira etapa desta pesquisa foi o levantamento bibliográfico, buscando organizar as referências. O desenvolvimento foi baseado nos materiais já produzidos sobre o assunto, como livros, artigos, dissertações, sites da internet, projetos e ou produtos comerciais e projetos livres já desenvolvidos. Fontes variadas, como sites específicos sobre Computação Gráfica, modelagem 3D, técnicas aplicadas em ambientes 3D e artigos científicos. Mas tanto no que diz respeito as atividades desenvolvidas: modelagem e renderização em 3D, pretendeu-se abordar uma bibliografia que fundamentasse o tema de forma acadêmica.

5.1.2 Escolha da ferramenta utilizada para confecção de modelos

Para a confecção do protótipo, foi escolhido à ferramenta Blender versão 2.63a que é o mais recente lançamento da Fundação Blender, criada por desenvolvedores voluntários. É importante frisar que o Blender é *Free & Open Source Software*, ou seja, é um software livre, gratuito, com o código fonte disponível sob a licença GNU GPL. A Blender Foundation na Holanda coordena o seu desenvolvimento contínuo.

Ele foi concebido em dezembro de 1993 e nasceu como um produto utilizável em agosto de 1994 como uma aplicação integrada que permite a criação de uma gama diversificada de conteúdos 2D e 3D. O Blender tem uma interface muito incomum, altamente otimizada para produção gráfica 3D, fornece um amplo espectro de modelagem, texturização, iluminação, animação e funcionalidade de vídeo de pós-processamento em um único pacote.

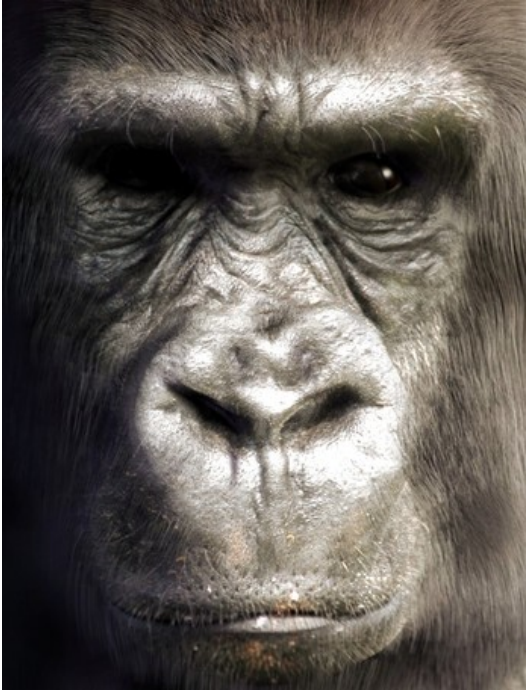
Por meio de sua arquitetura aberta, também fornece interoperabilidade entre plataformas, extensão e um fluxo de trabalho totalmente integrado. Blender é um dos mais populares *Open Source* aplicativos gráficos em 3D do mundo.

Principais Características:

- Oferece criação totalmente integrada, com uma ampla gama de ferramentas essenciais para a criação de conteúdo 3D, incluindo Modelagem, Mapeamento UV, Texturização, *Rigging*, *Skinning*, Animação, Simulação de Partículas entre outras, funciona com *Scripts*, e perfaz Renderização, Composição, Pós-produção, e criação de jogos;
- Multiplataforma, com interface gráfica OpenGL uniforme em todas as plataformas (personalizáveis com *scripts* em *Python*), pronta a ser utilizada nas versões mais correntemente utilizadas do Microsoft Windows, (Xp, Vista, Windows 7), do Linux, Mac OS X, FreeBSD, mas pode ser compilada para Irix, Sun e outros sistemas;
- Arquitetura 3D de alta qualidade que permita a criação de fluxo de trabalho rápido e eficiente;
- Mais de 200.000 *downloads* de cada lançamento (utilizadores) a nível mundial;
- Suporte ao usuário da comunidade por fóruns para perguntas, respostas e crítica na <http://BlenderArtists.org> e serviços de notícia na <http://BlenderNation.com>;
- O tamanho pequeno do executável, fácil distribuição.

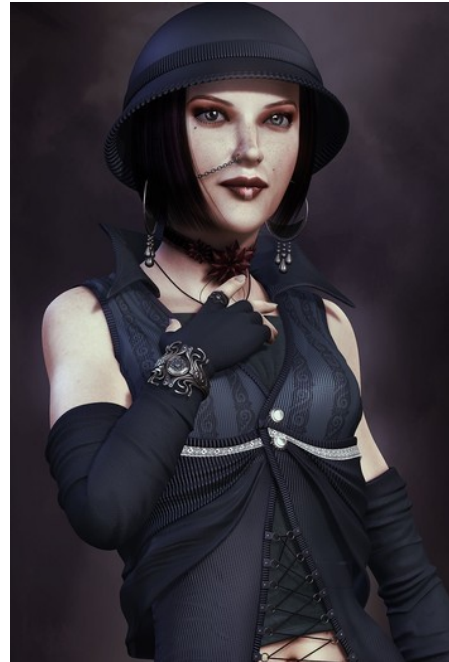
Essas e mais informações foram tiradas do site do Blender, assim como as Figuras 46, 47 e 48 que exemplificam de maneira mais complexo o realismo que essa ferramenta pode aplicar.

Figura 46 - Gorilla - Everett Gunther



Fonte: Blender

Figura 47 - Mulher com Chapéu - Zoltan Miklosi



Fonte: Blender

Figura 48 - Registe do Juggernaut - Derek Watts



Fonte: Blender

O Blender possui sua própria "*Game Engine*" embutida, que possibilita a criação de aplicações interativas 3D. A "*Game Engine*" do Blender ou a chamada "*BGE*" (*Blender Game Engine*) é uma poderosa ferramenta de programação de alto nível. Seu foco principal é o desenvolvimento de Jogos, mas a BGE pode ser utilizada para criar praticamente qualquer tipo de software interativo 3D para outras finalidades, como visualizações de cenários interativos 3D para arquitetura, ou então pesquisas físicas em educação. Abaixo, na Figura 49,

o jogo *Yofrankie*, onde o *Blender* foi utilizado para modelagem e animação, *Crystal Space* como engine 3D e plataforma de entrega, e para alguns scripts *Python* magia.

Figura 49 - Yofrankie



Fonte: Yofrankie

Esta ferramenta foi escolhida também por permitir facilmente a aplicação das técnicas estudadas, possuir uma grande comunidade ativa, ser utilizada em grandes projetos e ser gratuita. Apesar de ser uma ferramenta para confecção de modelos em 3D, como a nossa avaliação tem foco no Realismo Visual na etapa estática, na qual são tratados os objetos e cenas estáticas buscando o realismo fotográfico ou fotorrealismo, o protótipo em 2D também foi confeccionado nele. O que também permite uma comparação tratando-se de consumo mais precisa, visto que foi usado o mesmo software com as mesmas capacidades e características. Pode parecer curioso, mas as imagens nos mostraram como isso foi possível.

O modelo confeccionado é simples, para mero estudo e comparação de custos, longe da capacidade de impressão de realismo que essa ferramenta oferece.

5.1.3 Prototipação: realismo visual e textura

Cada vez mais se integra o Realismo aos jogos computacionais, muitas vezes esse é um quesito imprescindível para alguns títulos. Para adquirir uma visualização realista, há exigência de múltiplos recursos e, notoriamente, desenvolvendo-se apenas modelos tridimensionais, aplicando-se iluminação e coloração ainda assim não se consegue resultados satisfatórios. O que determina que algo a mais deve ser aplicado à imagens.

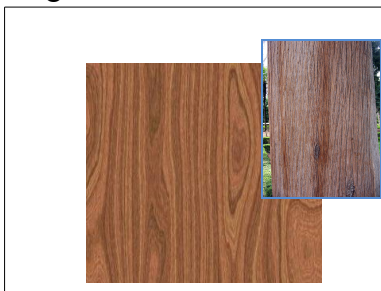
Então entram em cena as texturas, que são imagens aplicadas sobre modelos,

frisando que os mesmos podem ser arredondados, curvilíneos e não somente planos, elas representam materiais que compõem os modelos, como metal, concreto, madeira, pele, entre outros, elas dão atributos mais vivos para ambientes, objetos e personagens. As texturas colaboram na percepção de movimento, profundidade das composições de cenas.

Utilizando-se dos estudos, para o desenvolvimento do protótipo foram confeccionados: uma mesa e uma cesta de frutas para a demonstração das técnicas e processos envolvidos na geração do realismo visual. Esses objetos foram escolhidos para compor o modelo por exporem vários tipos de texturas. O objetivo é poder de uma maneira que seja simples e mais nítida analisar a aplicação de textura e emprego de técnicas que deem mais realismo a ela.

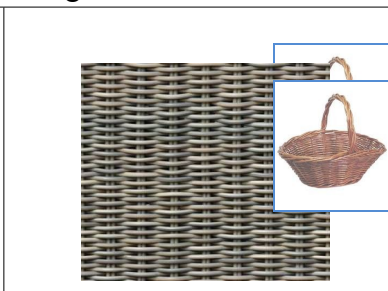
No modelo desenvolvido o objetivo foi aplicar as texturas e renderizar por passadas, pois geraria: economia de memória, uma vez que nem todos os objetos são colocados de uma vez para a renderização; a facilidade da introdução de modificações; maior utilização das imagens estáticas; e mixagem com objetos ou texturas reais obtidas por captura possibilitando integração com imagens fotográficas, acrescentando sombras, texturas complexas ou elementos do mundo real (como as texturas das Figuras 50, 51, 52, utilizadas na confecção do modelo) à cena. Salienta-se que qualquer textura aplicada em modelos 3D é uma imagem bidimensional. A representação de cenas tridimensionais é, naturalmente, bem mais complexa que a de figuras bidimensionais, mas elas também podem usar os mesmos conceitos de símbolos e hierarquias das figuras bidimensionais.

Figura 50 – Textura madeira



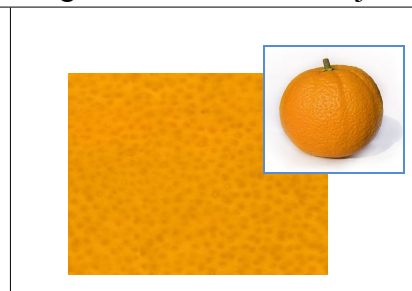
Fonte: CGTEXTURES

Figura 51 – Textura cesta



Fonte: CGTEXTURES

Figura 52 – Textura laranja

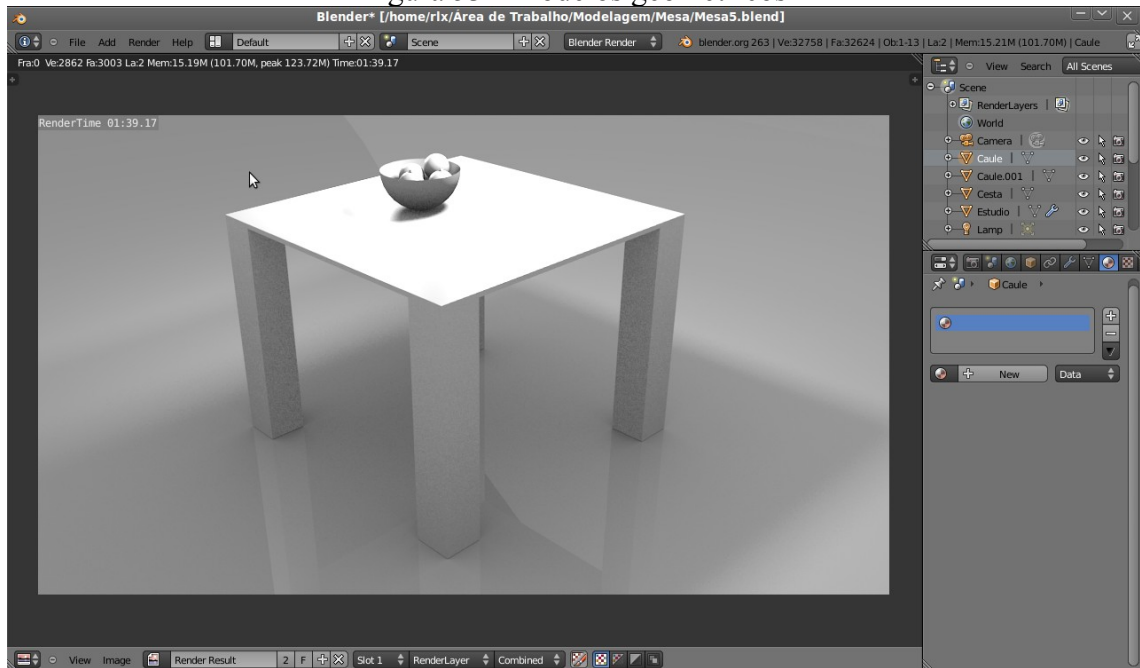


Fonte: CGTEXTURES

Esse modelo tridimensional apresenta uma vista em perspectiva dos objetos como se pode observar a seguir. A composição do modelo se deu por etapas:

- Na primeira foram confeccionadas modelos geométricos sólidos, ilustrados na Figura 53:

Figura 53 - Modelos geométricos



Fonte: Do autor (2012)

- Após a criação de modelos e a disposição dos mesmos no ambiente, aplicaram-se as texturas (Figura 54):

Figura 54 - Texturização



Fonte: Do autor (2012)

- Na sequencia da texturização aplicou-se o *bump map*, conforme Figura 55:

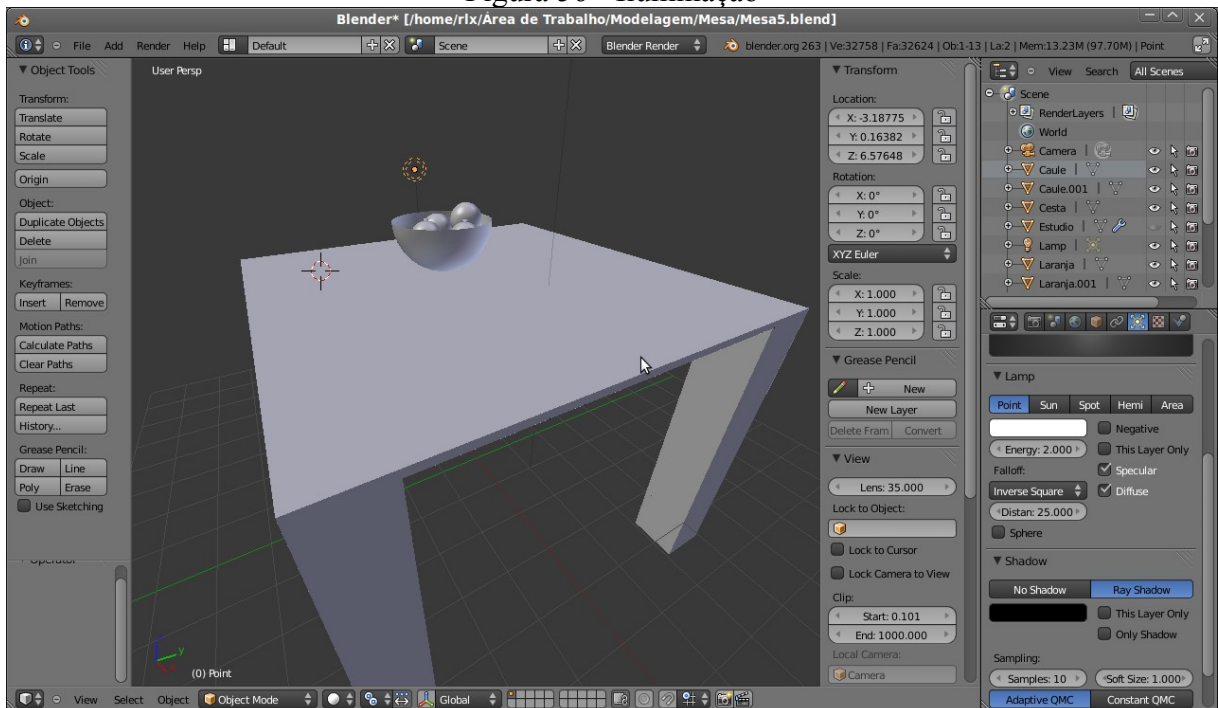
Figura 55 - Aplicação de *Bump map*



Fonte: Do autor (2012)

- Como podemos observar na Figura 56, o próximo passo foi ajustar a iluminação do tipo point light:

Figura 56 - Iluminação



Fonte: Do autor (2012)

- Última etapa visualizar a imagem renderizada (Figura 57):

Figura 57 - Finalização



Fonte: Do autor (2012)

- Na próxima Figura, 58, a imagem em 3D finalizada, vista de cima:

Figura 58 - Finalização, de cima e 3D



Fonte: Do autor (2012)

O processo de renderização envolveu todos essas fases como o previsto e estudado:

- a) construção do modelo;

- b) transformações lineares;
- c) eliminação de polígonos ou faces escondidas (*culling back faces*) – o próprio *Blender* executa essa fase;
- d) recortes (*clipping*) – o próprio *Blender* executa essa fase;
- e) rasterização – o próprio *Blender* executa essa fase;
- f) tratamento de partes escondidas (*hidden*) – o próprio *Blender* executa essa fase;
- g) coloração, textura e iluminação.

Os recursos utilizados para a confecção desse modelo foram os mesmos do em 2D:

- *Hardware*: Microcomputador Padrão PC, com os requisitos: processador Intel Core 2 Duo 2 Ghz T7300; 4 GB DDR2 de memória RAM; 250 GB HD; Intel GMA X3100, placa de vídeo. Todos os equipamentos necessários foram viabilizados pelo acadêmico.
- *Software*: Linguagens de programação e ferramentas para desenvolvimento. Os produtos estão disponíveis, de forma gratuita, pela empresa provedora sob licença freeware.
- *Peopleware*: Pessoas que conhecem ou trabalham na área. Professores e colegas.

5.1.4 Análise do protótipo

Tabela 2 – Análise do protótipo

ANÁLISE PROTÓTIPO	
Itens	Modelo 3D
<i>Software</i>	BLENDER VERSÃO 2.63A
<i>Hardware</i>	Microcomputador Padrão PC, com os requisitos: processador Intel Core 2 Duo 2 Ghz T7300; 4 GB DDR2 de memória RAM; 250 GB HD; Intel GMA X3100, placa de vídeo.
Texturas	Madeira: 450x297px; formato JPEG; tamanho 19,9KB. Cesta: 359x340px; formato JPEG; tamanho 28,1KB. Laranja: 340x340px; formato JPEG; tamanho 7,7KB.
Arquivo fonte	2928 polígonos; 865,9KB
Arquivo gerado	960x540px; formato PNG; 310KB
Técnicas aplicadas	Iluminação Point; texturização; bump-map.
Imagem	Figura 57
Tempo de renderização	11.15s (somente os objetos, sem cenário) 01:46.21s (com cenário, chão refletivo e sombra da mesa)

Fonte: Do autor (2012)

5.2 DISCUSSÃO E ANÁLISE DOS RESULTADOS OBTIDOS

Com a pesquisa e desenvolvimento do trabalho pode-se analisar que as técnicas de aplicação de texturas, que ao passar do tempo, tornaram-se cada vez mais aperfeiçoadas, para a evolução do Realismo Visual, um bom exemplo pode ser visto nos jogos digitais, o vídeo *Evolução do Realismo nos Gráficos*, apresentado na banca do autor, dia 26 de junho de 2012, demonstra isso. É importante salientar que aprimoramento do Realismo só foi possibilitado pela evolução dos *hardwares* disponíveis - consoles de última geração e computadores poderosos - a quem cria, desenvolve e joga e ou assiste os jogos digitais, animações, entre outros, profissionalmente ou não. Esses *hardwares* permitem a criação de elaborados e sensíveis ambientes de ficção primorosos que integram o jogador de forma mais intensa nesses mundos virtuais.

Notoriamente as novas tecnologias podem ser utilizadas com uma facilidade crescente. Uma textura bem desenvolvida é indispensável para aproximar cada vez mais os jogos a um maior patamar de qualidade, uma vez que, o que torna isso admissível é a sensação que o jogador tem de aproximar-se de um móvel como a mesa de madeira, ou a cesta

de palha do modelo desenvolvido no presente trabalho e visualizar o realismo da textura nela empregada.

Mas todo o realismo que o uso de técnicas de texturas e outras geram um custo computacional alto, uma grande demanda de processamento é necessário, exigindo super máquinas, dependendo da complexidade, para conseguir o efeito desejado. Esse é um dos problemas que mais limita o avanço da fidelidade com o real em ambientes digitais. Uma das soluções aplicáveis a fim da redução da complexidade de processamento é o acréscimo de detalhes na geometria da superfície dos modelos e a utilização de mapas de textura a partir de uma função ou mapeamento bidimensional.

A técnica de *Bump Mapping* é umas das técnicas de mapeamento de texturas e ela foi aplicada no modelo desenvolvido para avaliação desse Trabalho de Conclusão de Curso. Essa técnica é empregada para acrescentar realismo sem alteração da geometria do objeto, ela adiciona um sombreamento nos pixels, produzindo uma ilusão de relevo no objeto renderizado. Ela perturba o vetor normal em vários pontos da superfície criando uma ilusão de que algumas partes da superfície estariam elevadas ou rebaixadas, fazendo com que zonas de maior intensidade de cor (branco) pareçam em alto-relevo e as de menor intensidade (preto) pareçam em baixo-relevo, ou vice-versa.

Indica-se à quem está começando e ou aprendendo, ou se interessa em desenvolvimento de ambientes virtuais realísticos, que é interessante fazer uso de mapas de texturas; e do *Bump Mapping*, sempre que o mapa de textura for mais rico em diferenças de intensidade de cores. Frisando que quanto menor as diferenças de intensidades de cores no mapa de textura, menor será a ilusão de relevo. Logo, um mapa pobre em diferença de intensidade de cores, apresentará pouca percepção de relevo, portanto, dependendo do resultado que se busca, a *Bump Mapping*, não seria a técnica mais adequada, mas sim outras.

Sendo que os mapas de textura diminuem a complexidade de processamento, sugere-se para trabalhos futuros, um aprofundamento sobre essa questão e, já que eles podem ser aplicados através de várias técnicas, que as mesmas fossem aplicadas a um mesmo modelo com o mesmo mapa de textura, para que se possa fazer uma comparação e uma avaliação de quais técnicas de mapeamento de textura são mais eficientes, no aspecto de realidade visual, mas também de menor custo computacional. Para facilitar a escolha da técnica ideal para cada caso.

6 CONCLUSÃO

O realismo visual nos jogos computacionais da contemporaneidade é um dos requisitos fundamentais e indicador forte de qualidade que influencia na escolha e opinião dos usuários, ou seja, é outro apelo a mais ao público, além da história em si do jogo. As imagens tonaram-se mais complexas, adicionaram-se de forma gradativa atributos referentes ao fotorrealismo. Os padrões atuais da indústria estão requisitando bastante a parte de CG referente à modelagem de ambientes hiperrealistas para diversos setores, além dos jogos e animações, mas também, por exemplo, em maquetes eletrônicas de edifícios, residências. Mesmo que não esteja disponível a imersão total, na qual não há limite de interatividade e diferença entre o real e virtual, muito já se avançou nesse sentido.

Este trabalho *REALISMO VISUAL: RENDERIZAÇÃO DE IMAGENS EM JOGOS COMPUTACIONAIS* buscou chamar atenção, de leigos, mas também e principalmente de adoradores dos jogos computacionais e dos que desejam serem produtores dos mesmos, do qual esse conceito é interessante para o mercado, mesmo que seja complexo realizá-lo; ao mesmo tempo é um incentivo, mostrando as variadas técnicas - focamos neste trabalho as de texturas - que auxiliam na visualização realista de cenários, personagens, objetos, natureza e que cada vez mais se criam *softwares* e *hardwares*, que facilitam todo esse processo.

Se faz importante também ressaltar que a interatividade, que os jogos computacionais proporcionam é abordada como uma tecnologia que evolui firmemente, tornando-se um diferencial comparado as demais mídias, modificando não apenas o modo de jogar, mas o modo de visualização e percepção de quem joga.

Espera-se que esse presente trabalho, tenha podido apresentar de forma simples a evolução do realismo nos jogos, sua importância no mercado atual, os processos de desenvolvimento dos jogos computacionais, especificamente a renderização por passadas e as técnicas de texturas. Essa apresentação buscou despertar interesse para a ampliação de pesquisas e incentivos para estudos e aplicações práticas nesse seguimento. Quem ganha com essa evolução do realismo em jogos não são somente os que já são fãs e já tem seus jogos favoritos adequados a esse conceito, mas também o público que se interessa por um modo de entretenimento diferenciado, interativo e mais envolvente, a ciência que desenvolve mais e mais as tecnologias nesse meio, os profissionais que buscam ambientes virtuais mais afinados com a realidade e o mercado que vende esse produto continuamente oferecendo inovações.

REFERÊNCIAS

- ALMEIDA, Marcos Romero Gonzaga de. **Programação de jogos 2D usando o MRDX e Fundamentos de Jogos 3d**. Universidade Federal do Pará - Centro de Ciências Exatas e Naturais. 2004.
- ANDRADE, Rudi Lopes Bravo de. **Um Estudo de Técnicas Realistas de Renderização em Tempo Real**. 2007. Relatório do Trabalho de conclusão de curso (Ciência da computação). Universidade Federal de Santa Catarina, Florianópolis – SC.
- AUMONT, J. **A imagem**. São Paulo: Papirus, 1995.
- AUTODESK. WIKIHELP. Disponível em: http://wikihelp.autodesk.com/Revit/ptb/2012/Help/0000-Guia_do_01/2921-Personal2921/2931-Configur2931/2949-Materiai2949/2965-Alterand2965/2974-Mapas_pr2974/2979-Ru%C3%ADdo2979>. Acessado em: 11 jun. 2012.
- AZEVEDO, Eduardo; CONCI, Aura. **Computação gráfica: geração de imagens**. São Paulo: Campus, 2003.
- AZEVEDO, Eduardo; CONCI, Aura; R. LETA, Fabiana. **Computação Gráfica - Volume 2**. Rio de Janeiro: Editora Campus, 2008.
- BATES, Bob. **Game Design: The Art & Business of Creating Games**. 2ª Ed. Estados Unidos: Thomson Course Technology, 2004. 354p.
- BATTLEFIELD. Produtoras EA Games e DICE. Disponível em: <http://www.walldesk.com.br/papel-de-parede/wallpapers-games-battlefield.asp?f=5562>>. Acesso em: 14 nov. 2011.
- BLEND IT!. Disponível em: <http://blendit.wordpress.com/category/textura/>> Acessado em: 09 de jun. 2012.
- BOLTER, Jay David. **Remediation : understanding new media**. Cambridge (MA) : MIT, 2000. 295 p.
- BROWN, Harry J. **Videogames and education**. Reino Unido: M.E. Sharpe, 2008.
- CASTRO FILHO, Rômulo José de Lucena . **Construção de um modelador de curvas e superfícies de b ezier** . 2007. Graduação em Ciência da Computação. Faculdade Farias Brito, Fortaleza – CE.
- CECIL. Produtora Square-Enix. Disponível em: <http://www.ff-finalfantasy.com/ffiv-official-wallpaper10-1280x1024.html>>. Acesso em: 14 nov. 2011.
- CGTEXTURES. Disponível em: <http://www.cgtextures.com>>. Acesso em: 08 jul. 2011.
- CHONG, Andrew. **Animación Digital**. Barcelona: Blume, 2010. 175 p.

COMPUTER ARTS BRASIL. Edição nº 3. São Paulo: Editora Europa, 2007.

COUCHOT, Edmond. **A Tecnologia na Arte: da Fotografia a Realidade Virtual**. Porto Alegre: Ed. Da UFRGS, 2003. (Coleção Interfaces).

CORRÊA, Renata; GOMES, Silmara Pedretti. **Mip-mapping**. Monografia UNICAMP - Universidade Estadual de Campinas, 2004. Disponível em: <http://www.dca.fee.unicamp.br/courses/IA725/1s2004/monografias/MipMap_slides.pdf>. Acesso em: 14 nov. 2011.

CRYTEK 01. Disponível em: <http://crytek.com/sites/default/files/gallery_images/img-ce3-02.jpg>. Acesso em: 14 nov. 2011.

CRYTEK 02. Disponível em: <http://crytek.com/sites/default/files/gallery_images/img-ce3-21.jpg> Acesso em: 14 nov. 2011.

DALMAU, D. S.-C. **Core Techniques and Algorithms in Game Programming**. Editora: New Riders Publishing. 2003.

DOOM. Produtora id Software. Disponível em: <<http://3.bp.blogspot.com/-NFtY0mcDVzQ/TheXtZYYQUI/AAAAAAAAAAuY/FdB9rMH4ODY/s1600/Doom+imagem+4.jpg>>. Acesso em: 14 nov. 2011.

FACON, Jacques. **Processamento e análise de imagens**. Embalse: EBAI, 1993.

FOLEY, James D. et al. **Computer Graphics: Principles and Practices, Second Edition in C**. Massachusetts: Addison-wesley, 1996.

GONZALEZ, R., WOODS, R. **Processamento de Imagens Digitais**. São Paulo: Editora Edgard Blücher, primeira ed., 2000.

HEAVENLY. Produtora Ninja Theory. Disponível em: <http://2.bp.blogspot.com/___cgcmntNG6bo/S6zJg22gd_I/AAAAAAAAAqo/6WCDCI6bAvw/s320/heavenly-sword-desktop-1165.jpg>. Acesso em: 14 nov. 2011.

HDNLE.COM. Disponível em: <<http://www.hdnle.com/serv03.htm>>. Acesso em: 09 de jun. 2012.

HUTCHINSON, Andrew. **Making the Water Move: Techno-Historic Limits in the Game Aesthetics of Myst and Doom**. 2008. Disponível em: <<http://gamestudies.org/0801/articles/hutch>>. Acesso em: 20 de out. de 2011.

JENISCH, Josh. **The Art of the Videogame**. Estados Unidos: Quirk Books, 2008.

JONES, Wendy. **Beginning DirectX 9**. Boston: Thomson, 2004.

KOVACH, Peter J. **Inside Direct3D**. Redmond, Washington: Microsoft Press. 2000.

LAZZARI, Rafael. **Trabalho de Conclusão de Curso: Magic Bracelet**. Trabalho de conclusão de curso (Técnico em desenvolvimento de software). Escola Técnica de Farroupilha / Universidade de Caxias do Sul. Farroupilha – RS, 2005. 35p.

LIESER, Wolf. **Arte Digital: Novos Caminhos na Arte**. Alemanha: Tandem Verlag GmbH, 2010. 280 p.

LIMA, Edirlei Everson Eoares De. **3D Game Builder: Uma Game Engine Para a Criação de Jogos 3D**. 2008. Trabalho de conclusão de curso (Ciência da computação). Universidade do Contestado – UNC. Porto União - SC.

LION KING. Produtoras Disney e Virgin, licenciadas pela Nintendo. Disponibilizada pelo autor.

LUZ, Raphael. Disponível em: <<http://antigo.alphachannel.com.br/imagens/foto7.jpg>>. Acesso em: 07 nov. 2011.

MARQUES FILHO, Ogê; VIEIRA NETO, Hugo. **Processamento Digital de Imagens**, Rio de Janeiro: Brasport, 1999. ISBN 8574520098.

MARSHALL, D. **What is Multimedia?** Disponível em: <<http://www.cs.cf.ac.uk/Dave/Multimedia/node10.html>>. Acesso em: 02 nov. 2011.

MARVEL. Produtoras Marvel e Capcom. **Marvel vs Capcom 3: Fate of Two Worlds** – 2011.

MEDAL OF HONOR. Produtoras EA Games e Nintendo. Disponível em: <[http://www.gamesbasement.co.uk/userfiles/image/Medal%20of%20Honor%20Heroes%20%20\(1\).jpg](http://www.gamesbasement.co.uk/userfiles/image/Medal%20of%20Honor%20Heroes%20%20(1).jpg)>. Acesso em: 14 nov. 2011.

MEGAMAN. Produtora Capcom. Disponível em: <<http://www.themnetwork.com/gallery/displayimage.php?album=152&pid=2594>>. Acesso em: 14 nov. 2011.

MIGUEL, G. Cesar; MONTEIRO, L., R., Júlio; CAVALHIERI, A., Marcos. **Componentes de uma Game Engine**. 2006.

MORRIS, Dave. HARTAS, Leo. **The Art of Game Worlds**. Estados Unidos: HarperCollins, 2004. 192 p.

MÜLLER, Eduardo Fernando. **Os Conceitos Estético-Visuais dos Jogos Digitais**. Dissertação de Mestrado em Comunicação Social para obtenção de título de Mestre. Pontifícia Universidade Católica do Rio Grande do Sul. 2011.

NEGROPONTE, N. **A vida digital**. São Paulo: Cia das Letras, 1995.

OPENGL Architecture Review Board. Disponível em: <<http://www.opengl.org>>. Acesso em: 02 out. 2011.

PAULA FILHO, Wilson de Pádua. **Multimídia: conceitos e aplicações**. Rio de Janeiro: LTC, c2000. 312 p.

PERSIANO, R. C. M. and OLIVEIRA, A. A. F. **Introdução a Computação Gráfica**. Livros Técnicos e Científicos. Editora Ltda, 1989

PINHEIRO, Cristiano Max Pereira. **Apontamentos para uma aproximação entre jogos digitais e comunicação**. Porto Alegre, 2007. 201p.

PROJECTALL. Disponível em: <<http://projectall.wordpress.com/>>. Acesso em: 09 de jun. 2012.

RÉNAUD , Alain. **"Pensare l'Immagine Oggi. Nuove Immagini, Nuovo Regime del Visibile, Nuovo Immaginario"**. In V.A., Videoculture di Fine Secolo. Napoli, Liguori, 1989, pp. 11-27.

ROLLINGS, Andrew; MORRIS, Dave. **Game Architecture and Design**. Arizona: Coriolis, 2000.

SILENT HILL. Produtora Konami. Disponível em: <<http://www.juegomania.org/Silent+Hill:+Homecoming/foto/ps3/0/221/10.jpg/Foto+Silent+Hill:+Homecoming.jpg>>. Acesso em: 14 nov. 2011.

SILVA, Haroldo Watson Teodósio da. **Sombreamento realista para ambientes virtuais**. 2010. 39p. Trabalho de conclusão de curso (Ciência da computação). Universidade Federal do Rio Grande do Norte, Natal – RN.

SILVEIRA NETO, Walter Dutra da; MELO , Andrei Krepsky de. **Técnicas de animação em ambientes 3D** . Disponível em: <http://www.ceart.udesc.br/revista_dapesquisa/volume1/numero2/design/animacao_ambientes_3D.pdf>. Acesso em: 02 out. 2011.

TATSUNOKO. Produtoras Tatsunoko e Capcom. Tatsunoko vs. Capcom: Ultimate All-Stars – 2009.

TAVINOR, Grant. The Art of Videogames. Estados Unidos: Blacwell Publishing, 2009.

TEXAS TECH COLLEGE OF ARCHITECTURE. Disponível em: <http://www.arch.ttu.edu/people/faculty/key_b/VIZ_NOTES/UVW_Mapping.htm> Acessado em: 10 jun. 2012.

TOP GEAR. Produtoras Kemco e Gremlin Graphics, licenciadas pela Nintendo. Disponibilizada pelo autor.

TORRES, Fabio. **Evolução de gráficos psicodélicos em jogos de luta 2D**. Disponível em: <<http://www.arkade.com.br/especiais/evolucao-graficos-psicodelicos-jogos-luta-2d>>. Acesso em: 14 nov. 2011.

VALCAR.NET. Disponível em: <<http://www2.dem.inpe.br/val/homepage/tutor/index.html>> Acessado em: 11 jun. 2012.

WEINFURTER, Delcio Vill. **Customização para acrescentar atributos em Sólidos do Solidworks**. 2003. Graduação em Ciência da Computação. FURB, Blumenau – SC.

WINCKLER, R. d. B.; MARTINS, A. **Jogos Educativos aplicados a e-Learning: mudando a maneira de avaliar o aluno**. 2003. Universidade Federal de Santa Catarina – UFSC.

WOLFENSTEIN 3D. Produtora id Software. Disponível em:
<http://www.abandonia.com/files/games/16/Wolfenstein%203D_5.png>. Acesso em: 14 nov. 2011.

YOFRANKIE. Disponível em: <<http://www.yofrankie.org>> . Acesso em: 10 maio 2012.

ZELDA. Produtora Nintendo. Disponível em:
<<http://totaldarkgames.blogspot.com/2011/07/25-aniversario-de-legend-of-zelda.html>>.
Acesso em: 14 nov. 2011.

ZERBST, Stefan; DÜVEL, Oliver. **3D Game Engine Programming**. Boston: Thomson, 2004.

APÊNDICE(S)

APÊNDICE A – Artigo

REALISMO VISUAL: RENDERIZAÇÃO DE IMAGENS EM JOGOS COMPUTACIONAIS**Rafael Lazzari¹**

¹Curso de Ciência da Computação
Universidade do Extremo Sul Catarinense (UNESC) Criciúma – SC – Brazil

rlx.exe@gmail.com

***Abstract.** This article presents the processes, stages of rendering, as well as the techniques used to obtain the visual realism in order to explain the reason he is a current trend in computer graphics and probably become permanent.*

***Resumo.** Este artigo apresenta os processos, as fases de renderização, assim como as técnicas utilizadas para obtenção do realismo visual, de maneira a explicitar o motivo de ele ser uma tendência na computação gráfica atual e provavelmente se tornar algo permanente.*

Palavras-chave: Realismo Visual; Computação Gráfica; Jogos Computacionais; Imagens Digitais; 2D; 3D; Blender.

1. Histórico

Um dos principais objetivos da Computação Gráfica (CG) é construir modelos para geração de imagens sintéticas realistas. O Realismo Visual, em CG, pode ser considerado em duas etapas: a estática, foco do TCC presente, na qual são tratados os objetos e cenas estáticas buscando o realismo fotográfico ou fotorrealismo, e a dinâmica que está relacionada ao movimento da cena e seus personagens focando a animação computacional.

2. Introdução

O Realismo Visual é sinônimo de boa qualidade e é fundamental e perceptível em aplicações como simulação, cenários e personagens de filmes e jogos, realidade virtual, criação de ambientes virtuais e reconstruções realísticas tridimensionais. Para obtê-lo, fazem-se modelos físicos cada vez mais sofisticados, que vêm sendo utilizados para modelar as interações das fontes de luz com os objetos. Outras evoluções podem ser observadas nas formas de representação geométrica (bi e tridimensional - 2D e 3D) e apresentação dos materiais que caracterizam uma imagem. Quanto mais sofisticado o modelo e as representações, mais demanda há de recursos tanto de software quanto hardware para síntese e renderização dessas imagens.

O realismo visual em jogos, observada na qualidade dos gráficos apresentados, é um dos requisitos fundamentais e indicador forte de qualidade que influencia na escolha e opinião dos usuários. Que grau de realismo será possível atingir em jogo computacional? Ainda que não

esteja disponível a imersão total, na qual não há limite de interatividade e diferença entre o real e virtual, muito já se avançou nesse sentido. Porém, um das preocupações do segmento de jogos de computador é a grande demanda de processamento necessária à renderização de imagens hiper realistas. Este é considerado um dos gargalos da indústria de jogos.

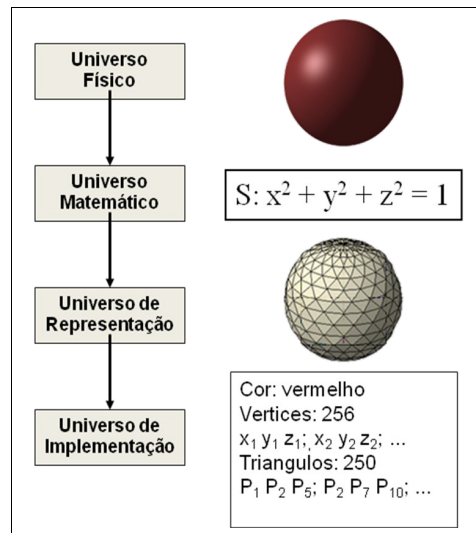
3. Computação gráfica e realismo visual em imagens 2D e 3D

De acordo com a *International Standards Organization (ISO)*, a Computação Gráfica (CG) é a área da Ciência da Computação que estuda a geração, manipulação e interpretação de modelos e imagens de objetos incluindo técnicas e métodos computacionais que convertem dados para dispositivos gráficos e vice-versa. Neste contexto, As imagens constituem uma representação visual de objetos de duas ou três dimensões descritos por meio de especificações abstratas. De acordo com Gomes e Velho (1998), a imagem é a materialização de grande parte dos processos da CG sendo que está presente em todas as suas áreas, seja como produto final (visualização), ou como parte do processo de interação (modelagem). A CG engloba assim, a síntese, o processamento e a análise de imagens (PERSIANO, OLIVEIRA, 1989).

3.1. Imagem digital (ID)

De acordo com Azevedo e Conci (2003), um objeto gráfico é representado por meio da especificação de seus atributos: geométricos (estabelecem as propriedades métricas dos objetos); topológicos (tratam da forma dos objetos), de cor (definem a informação de textura dos objetos). Para representar o mundo real no mundo virtual, usa-se na CG o paradigma de abstração: de níveis (contínuo e discreto) ou dos quatro universos (Físico, Matemático, Representação, Implementação) considerando os diferentes níveis e processos na representação de uma imagem na visão computação ilustrado na Figura 1 abaixo.

Figura 1 - Os quatro universos na abstração de uma esfera



De acordo com Toledo (2011), no Universo Físico, as imagens reais possuem um número ilimitado de cores ou tons.

3.2. Representação espacial: imagem 2D e 3D

A percepção de “espacialidade” de uma imagem pode ser vista como a capacidade que o ser humano tem de distinguir a forma, as cores, a textura e a relação espacial existente entre os

objetos. O mundo real é formado por um espaço fisicamente tridimensional sendo que a dimensão é um parâmetro ou medida requerida para definir as características de um objeto (por exemplo, largura, altura, profundidade ou tamanho e forma). Na matemática, a dimensão de um espaço é o número de parâmetros necessários para identificar um ponto e descrever sua posição ou objeto no espaço conceitual.

3.3. Técnicas de computação gráfica e realismo visual

Para Azevedo e Conci (2003), o objetivo final da CG é a criação de imagens sintéticas realistas. Sintetizar uma imagem (uma cena ou um objeto) é gerá-la considerando a definição dos dados dos objetos que a compõe. A qualidade de realismo em uma imagem está associada ao seu grau de convencimento, isto é, qual próxima está sua representação da realidade. O realismo visual em CG é composto por técnicas de tratamento computacional aplicada aos objetos gerados (por modelagem de sólidos, partículas, fractais ou outra técnica) com objetivo de criar-lhes uma imagem sintética o mais próxima da realidade com alto grau de verossimilhança como se tivessem sido construídos, fotografados ou filmados. (PAULA FILHO, 2000).

3.4. Realismo e o modelo

A Modelagem Geométrica, também chamada de geometria computacional, atua no campo da modelagem de sólidos, atrelando geometria com computação. Para realizar a modelagem geométrica são utilizados métodos que visam descrever a forma e as características geométricas de um objeto. Ela provê uma descrição ou modelo muito mais analítico, matemático e abstrato que o real.

4. Jogos digitais

Os Jogos Digitais (JD) são provavelmente a forma de arte mais avançada até agora na história humana, porque eles sintetizam texto, imagem, som, vídeo, e a participação ativa da audiência em uma experiência estética unificada (BROWN, 2008).

Eles abrangem diferentes tipos de recursos, que podem ser observados nos diversos exemplares existentes. Nesse capítulo explanaremos sobre conceitos, tipos, exemplos, mas também sobre a evolução dos jogos computacionais, no que diz respeito ao realismo visual.

4.1. Contextos e características de JD

Os jogos fazem parte do nosso cotidiano e de modo geral, os utilizamos de várias formas: os dos sentidos, em que a curiosidade nos leva ao conhecimento; os jogos corporais expressos na dança, nas cerimônias e rituais de certos povos; o jogo das cores, da forma e dos sons, presente na arte dos imortais; o jogo do olhar. Enfim, ele está presente em diversos lugares. A intensidade do poder do jogo é tão grande que nenhuma ciência conseguiu explicar a fascinação que ele exerce sobre as pessoas (WINCKLER, 2003).

Uma das classificações de tipos de jogos são a dos dois grandes grupos 2D e 3D, e o grupo de jogos que utiliza as duas tecnologias.

Figura 2 – Jogo 2D



Figura 3 – Jogo 3D



Há uma imensa quantidade de tipos de jogos e a cada dia surgem novos sucessos, para todos os gostos e com utilização de diferentes técnicas, estéticas, gêneros, que vão evoluindo e sendo aperfeiçoadas. Esse fato é uma maravilha aos adeptos a esse tipo de entretenimento e um convite atraente aos que ainda não são.

4.2. Evolução do realismo nos gráficos

Realismo e estilização são puramente respostas às perguntas. Exemplo, se um designer se aproxima de um artista, com um jogo de estratégia de Guerra, baseado na Primeira Guerra

Mundial, o Hiper-Realismo seria uma opção a se considerar, já uma solução no estilo da *Disney* seria considerado cômico para o público em potencial, por não combinar com o tema do jogo e não trazer a seriedade que o mesmo pede. Porém, aqui é onde uma falsa dicotomia é revelada. Estilização não é relativa apenas a formas ou volumes, mas pode ser expressa também em tons e texturas, e no caso de um jogo de Guerra, esta seria uma opção viável (MORRIS, 2007).

O realismo nos gráficos começa a evoluir a partir da década de 1980, ocorre uma mudança na concepção da animação pelo surgimento das técnicas de modelagem em 3D. Tais condições possibilitaram tanto fornecer as informações que descrevem o movimento dos objetos, quanto para usar o computador para sintetizar as informações a partir de algoritmos (COUCHOT, 2003).

A qualidade do fotorrealismo dentro dos Jogos Digitais se desenvolve pela captação por parte do artista 3D de idéias de geração de imagens com base na fotografia, na pintura hiper-realista e na escultura. Combinando elementos destes campos das artes, estas imagens trazem uma nova luz à consideração da forma, com uma complexidade específica de produção e convencimento de que aquilo que é visto denota o real (LIESER, 2010).

4.3. Game engines

Uma *game engine* é o núcleo básico para a criação de um jogo e tem o objetivo de simplificar o processo de codificação, além de fornecer uma série de funcionalidades agregadas. Ela pode ser dividida em diversas sub-engines, como por exemplo, engine gráfica – que será conceituada- engine física, engine sonora, engine de inteligência artificial, etc. (MIGUEL, 2006).

5. Realismo Visual: renderização de imagens em jogos computacionais

Para uma avaliação mais precisa das técnicas de computação gráfica relacionadas a texturização em imagens, evidenciando a qualidade do realismo visual e custo computacional, objetivo do trabalho proposto, foram confeccionados modelos gráficos em 2D e 3D, neles foram empregadas técnicas e processos de renderização para a obtenção de características mais próximas do real.

A avaliação das técnicas de texturização na renderização de imagens para jogos computacionais se faz importante, visto que, o Realismo Visual é um quesito importante para o crescimento positivo do mercado da Computação Gráfica.

5.1. Metodologia

Dentro das linhas de pesquisa do curso de Ciência de Computação, esse projeto se insere na linha de Computação Gráfica, pois ela proporciona a criação de ambientes virtuais que com o auxílio de técnicas computacionais cada vez mais avançadas deixam estes ambientes extremamente realistas.

A metodologia aplicada no presente trabalho se dá em duas fases, sendo que a primeira concentrou-se em realizar o levantamento bibliográfico, feita através de artigos, livros e trabalhos acadêmicos.

A segunda fase foi prática, a execução da mesma se deu a partir dos conhecimentos absorvidos, na fase teórica. Ela foi composta pelos seguintes itens:

- a) Desenvolvimento do protótipo;
- b) Aplicaram-se transformações lineares no modelo;
- c) Identificaram-se, eliminaram-se ou inseriram-se polígonos ou faces ocultas no modelo;
- d) Realizaram-se recorte ou inserção de elementos no modelo;
- e) Converteram-se o modelo em pixels;
- f) Foram tratadas as partes ocultas do modelo;
- g) Foram aplicadas técnicas de coloração, textura e iluminação.;
- h) O protótipo foi renderizado a partir das técnicas aplicadas;
- i) Testou-se o protótipo e foram relatados os resultados;
- j) Para conclusão foi feita avaliação de técnicas de texturização na renderização do protótipo evidenciando a qualidade do realismo visual.

5.2. Escolha da ferramenta utilizada para confecção de modelos

Para a confecção do protótipo, foi escolhido a ferramenta Blender versão 2.63a que é o mais recente lançamento da Fundação Blender, criada por desenvolvedores voluntários. É importante frisar que o Blender é *Free & Open Source Software*, ou seja, é um software livre, gratuito, com o código fonte disponível sob a licença GNU GPL. A Blender Foundation na Holanda coordena o seu desenvolvimento contínuo.

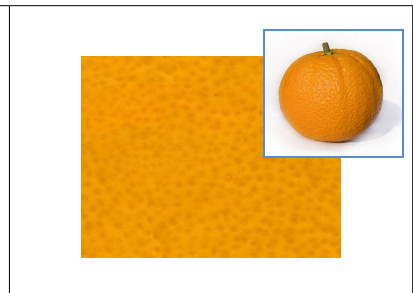
5.3. Prototipação: realismo visual e textura

Cada vez mais se integra o Realismo aos jogos computacionais, muitas vezes esse é um quesito imprescindível para alguns títulos. Para adquirir uma visualização realista, há exigência de múltiplos recursos e, notoriamente, desenvolvendo-se apenas modelos tridimensionais, aplicando-se iluminação e coloração ainda assim não se consegue resultados satisfatórios. O que determina que algo a mais deve ser aplicado à imagens.

Então entram em cena as texturas, que são imagens aplicadas sobre modelos, frisando que os mesmos podem ser arredondados, curvilíneos e não somente planos, elas representam materiais que compõem os modelos, como metal, concreto, madeira, pele, entre outros, elas dão atributos mais vivos para ambientes, objetos e personagens. As texturas colaboram na percepção de movimento, profundidade das composições de cenas.

Utilizando-se dos estudos, para o desenvolvimento do protótipo foram confeccionados: uma mesa e uma cesta de frutas para a demonstração das técnicas e processos envolvidos na geração do realismo visual. Esses objetos foram escolhidos para compor o modelo por exporem vários tipos de texturas. O objetivo é poder de uma maneira que seja simples e mais nítida analisar a aplicação de textura e emprego de técnicas que deem mais realismo a ela.

No modelo desenvolvido o objetivo foi aplicar as texturas e renderizar por passadas, pois geraria: economia de memória, uma vez que nem todos os objetos são colocados de uma vez para a renderização; a facilidade da introdução de modificações; maior utilização das imagens estáticas; e mixagem com objetos ou texturas reais obtidas por captura possibilitando integração com imagens fotográficas, acrescentando sombras, texturas complexas ou elementos do mundo real (como as texturas das Figuras 3, 4, 5, utilizadas na confecção do modelo) à cena. Salienta-se que qualquer textura aplicada em modelos 3D é uma imagem bidimensional. A representação de cenas tridimensionais é, naturalmente, bem mais complexa que a de figuras bidimensionais, mas elas também podem usar os mesmos conceitos de símbolos e hierarquias das figuras bidimensionais.

Figura 3 – Textura madeira**Figura 4 – Textura cesta****Figura 5 – Textura laranja**

Esse modelo tridimensional apresenta uma vista em perspectiva dos objetos como se pode observar na Figura 6 a seguir:

Figura 6 – Finalização

6. Conclusão

O realismo visual nos jogos computacionais da contemporaneidade é um dos requisitos fundamentais e indicador forte de qualidade que influencia na escolha e opinião dos usuários, ou seja, é outro apelo a mais ao público, além da história em si do jogo. As imagens tornaram-se mais complexas, adicionaram-se de forma gradativa atributos referentes ao fotorrealismo.

Se faz importante também ressaltar que a interatividade, que os jogos computacionais proporcionam é abordada como uma tecnologia que evolui firmemente, tornando-se um diferencial comparado as demais mídias, modificando não apenas o modo de jogar, mas o modo de visualização e percepção de quem joga.

Referências

AZEVEDO, Eduardo; CONCI, Aura. **Computação gráfica: geração de imagens**. São Paulo: Campus, 2003.

BROWN, Harry J. **Videogames and education**. Reino Unido: M.E. Sharpe, 2008.

COUCHOT, Edmond. **A Tecnologia na Arte: da Fotografia a Realidade Virtual**. Porto Alegre: Ed. Da UFRGS, 2003. (Coleção Interfaces).

LIESER, Wolf. **Arte Digital: Novos Caminhos na Arte**. Alemanha: Tandem Verlag GmbH, 2010. 280 p.

MIGUEL, G. Cesar; MONTEIRO, L., R., Júlio; CAVALHIERI, A., Marcos. **Componentes de uma Game Engine**. 2006.

MORRIS, Dave. HARTAS, Leo. **The Art of Game Worlds**. Estados Unidos: HarperCollins, 2004. 192 p.

PAULA FILHO, Wilson de Pádua. **Multimídia: conceitos e aplicações**. Rio de Janeiro: LTC, c2000. 312 p.

PERSIANO, R. C. M. and OLIVEIRA, A. A. F. **Introdução a Computação Gráfica**. Livros Técnicos e Científicos. Editora Ltda, 1989.

WINCKLER, R. d. B.; MARTINS, A. **Jogos Educativos aplicados a e-Learning: mudando a maneira de avaliar o aluno**. 2003. Universidade Federal de Santa Catarina – UFSC.