

UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

**MODELAGEM DE OBJETOS TRIDIMENSIONAIS PARA UM
AMBIENTE INTERATIVO DE INSTRUÇÕES TÉCNICAS VIRTUAIS**

MESSIAS JOSÉ AMADOR DO NASCIMENTO

UFPA / ITEC / PPGEE
Campus Universitário do Guamá
Belém - Pará - Brasil
2010

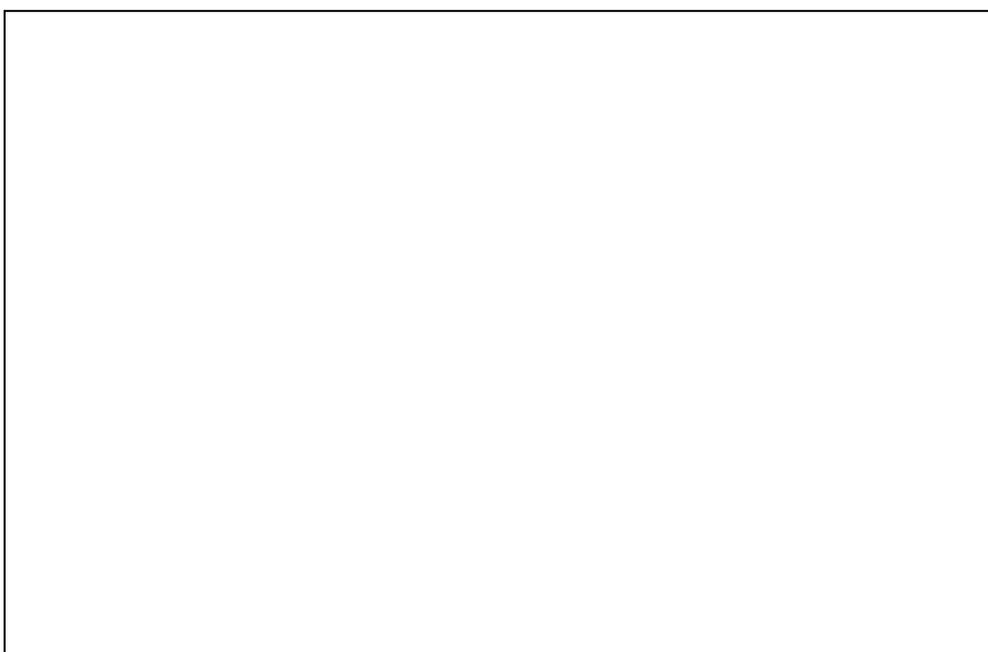
UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

**MODELAGEM DE OBJETOS TRIDIMENSIONAIS PARA UM
AMBIENTE INTERATIVO DE INSTRUÇÕES TÉCNICAS VIRTUAIS**

DISSERTAÇÃO SUBMETIDA À BANCA EXAMINADORA APROVADA PELO
COLEGIADO DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA
ELÉTRICA DO INSTITUTO DE TECNOLOGIA DA UNIVERSIDADE FEDERAL
DO PARÁ, COMO REQUISITO PARA A OBTENÇÃO DO GRAU DE MESTRE
EM ENGENHARIA ELÉTRICA NA ÁREA DE COMPUTAÇÃO APLICADA.

MESSIAS JOSÉ AMADOR DO NASCIMENTO

UFPA / ITEC / PPGEE
Campus Universitário do Guamá
Belém-Pará-Brasil
2010



**MODELAGEM DE OBJETOS TRIDIMENSIONAIS PARA UM AMBIENTE
INTERATIVO DE INSTRUÇÕES TÉCNICAS VIRTUAIS**

AUTOR: MESSIAS JOSÉ AMADOR DO NASCIMENTO

DISSERTAÇÃO DE MESTRADO SUBMETIDA À AVALIAÇÃO DA BANCA EXAMINADORA APROVADA PELO COLEGIADO DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA DA UNIVERSIDADE FEDERAL DO PARÁ, COMO REQUISITO PARA OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA ELÉTRICA NA ÁREA DE COMPUTAÇÃO APLICADA.

APROVADA EM: 10/09/2010

BANCA EXAMINADORA:

Prof. Dr. Manoel Ribeiro Filho
(ORIENTADOR – UFPA)

Prof. Dr. Marcus Vinicius Alves Nunes
(MEMBRO – UFPA)

Dr. Marcos Paulo Alves de Sousa
(MEMBRO – Museu Goeldi)

VISTO:

Prof. Dr. Marcus Vinicius Alves Nunes
(COORDENADOR DO PPGEE/ITEC/UFPA)

À minha esposa Lorena, com todo amor

AGRADECIMENTOS

Agradeço aos meus pais, Messias (*in memoriam*) e Maria Helena, por serem meus mais preciosos professores nessa vida, por todo amor e zelo, e por sempre darem o devido valor à educação de seus filhos. Aos meus irmãos, Messilena e Orlando, verdadeiros amigos e companheiros. Aos tios e tias que sempre torceram por mim. À minha esposa Lorena, por todo amor e apoio. Ao professor Manoel Ribeiro Filho, principalmente pela confiança, mas também pela paciência. A todos os colegas do Laboratório de Realidade Virtual - LaRV, com os quais tive a honra de trabalhar ao longo desses 5 anos. À Eletronorte, como instituição, pelo grande incentivo a este e outros projetos de pesquisa e tecnologia. Aos funcionários da Eletronorte, pela valorosa (e indispensável) colaboração com este trabalho. E a todos aqueles que indiretamente contribuíram ou simplesmente torceram pelo sucesso do sistema ITV. Um muitíssimo obrigado.

*“Deus contemplou toda a sua obra,
e viu que tudo era muito bom.”*

Gênesis, 1:31

RESUMO

Esta dissertação aborda metodologias de criação de objetos tridimensionais destinados a ambientes virtuais interativos, contextualizando as principais técnicas de implementação computacional envolvidas no processo. Foi utilizado como estudo de caso o Sistema de Instruções Técnicas Virtuais, desenvolvido pelo Laboratório de Realidade Virtual (LaRV) da Universidade Federal do Pará (UFPA) para as Centrais Elétricas do Norte do Brasil S/A (Eletronorte), como recurso de treinamento e simulação de procedimentos de manutenção e operação da Usina Hidrelétrica de Tucuruí.

PALAVRAS-CHAVE: Computação Gráfica, Ambientes Virtuais, Modelagem 3D, CAD

ABSTRACT

This dissertation discusses creation paradigms of tridimensional objects, to be used into interactive virtual environments, contextualizing the main techniques for computing implementation of the whole process. As a case study, it was used the *Virtual Technical Instructions System*, developed by *Virtual Reality Laboratory* at *Federal University of Pará*, to *Eletronorte S/A*, as an auxiliary resource of simulation and training in maintenance and operation procedures at *Tucuruí Hydroelectric Power Plant*.

KEYWORDS: Computer Graphics, Virtual Environments, 3D Modeling, CAD

SUMÁRIO

RESUMO.....	viii
ABSTRACT.....	ix
LISTA DE FIGURAS.....	xiii
Capítulo 1 – Introdução.....	1
Capítulo 2 – Trabalhos Correlatos.....	3
2.1 Unidade Geradora Virtual.....	3
2.2 Comparação entre ambientes 3D e outras mídias de treinamento.....	5
Capítulo 3 – Ambientes Virtuais 3D.....	6
3.1 Interação e Imersão.....	6
3.2 Interfaces.....	7
Capítulo 4 – Objetos 3D.....	9
4.1 Aplicações-alvo.....	9
4.2 Métodos de Representação Geométrica.....	12
4.2.1 Voxels.....	12
4.2.2 Curvas Parametrizadas.....	13
4.2.3 Polígonos.....	13
4.2.4 Sistemas de Partículas.....	14
4.3 Objetos realísticos e otimizados.....	15
4.3.1 Geometria.....	15
4.3.2 Textura.....	16
4.3.2.1 Mapeamento.....	18
4.3.2.2 Transparência.....	18
4.3.3 Material.....	19
4.3.3.1 Sub-materiais.....	20
4.3.3.2 Máscaras.....	22
4.3.3.3 Materiais para múltiplos objetos.....	23
4.4 Animações embutidas.....	24

4.5 Criação.....	25
4.5.1 Softwares modeladores 3D.....	26
4.5.2 Softwares de Edição de Imagens Bitmap.....	26
Capítulo 5 – Estudo de caso: Sistema de Instruções Técnicas Virtuais.....	28
5.1 Interface GUI e Ambiente 3D.....	28
5.2 Estrutura Funcional.....	29
5.3 Arquitetura Computacional.....	30
5.3.1 Linguagem C++.....	30
5.3.1.1 QT.....	31
5.3.1.2 Ogre3D.....	31
5.3.2 XML (eXtensible Markup Language).....	31
5.3.3 CAD (Computer-Aided Design).....	33
5.4 Organização da media.....	34
5.4.1 Arquivos utilizados pelo Ogre3D.....	34
5.4.1.1 Geometrias.....	34
5.4.1.2 Texturas.....	34
5.4.1.3 Materiais.....	35
5.4.1.4 Skeletons.....	35
5.4.2 Padrão de nomenclatura.....	35
5.4.2.1 Grupos de objetos.....	35
5.4.2.2 Nomes de entidades.....	36
5.4.3 Estrutura de diretórios.....	36
5.5 Contexto da Modelagem 3D.....	37
5.5.1 Cenário real: instalações da UHE Tucuruí.....	37
5.5.2 Recursos e Informações Disponíveis.....	38
5.5.3 Referências Métricas.....	39
5.6 Exemplos de objetos 3D criados.....	42
5.6.1 Terreno.....	42
5.6.2 Água.....	44
5.6.3 Concretos.....	45
5.6.4 Rotor do Gerador (versão simplificada).....	46
5.6.5 Tubulações com setas animadas.....	47
5.6.6 Grades.....	48

5.6.6.1	Grade da tomada d'água.....	49
5.6.6.2	Grades do teto.....	49
5.6.6.3	Grades de proteção.....	50
5.6.7	Manômetros e ponteiros.....	51
5.6.8	Centro de Operação da Usina.....	51
5.6.8.1	Telas dos monitores.....	52
5.6.8.2	Corredor.....	53
5.6.8.3	Vasos de plantas.....	53
5.6.9	Parafusos.....	54
5.6.10	Tampa Superior.....	54
5.6.11	Turbina Francis.....	55
5.6.12	Escotilha da caixa espiral.....	56
5.6.13	Molas.....	56
5.6.13.1	Válvula detectora de pressão.....	56
5.6.13.2	Sistema de frenagem.....	57
Capítulo 6 – Conclusão.....		58
Referências Bibliográficas.....		59

LISTA DE FIGURAS

Figura 2.1: Alguns cenários do sistema UGV.....	4
Figura 2.2: Aspectos visuais e estruturais dos termos 2D e 3D	5
Figura 2.3: Ambiente 3D para instruções de montagem e gráfico comparativo.....	5
Figura 3.1: Luvas para interação com ambientes virtuais.....	6
Figura 3.2: Controlador de jogos de corrida.....	7
Figura 3.3: Interface e ambiente 3D do software Google Earth.....	8
Figura 3.4: Ambiente 3D com controle de iluminação por meio de elementos GUI.....	8
Figura 4.1: Diferentes níveis de detalhamento de objetos 3D.....	11
Figura 4.2: Aspecto visual de um objeto formado por voxels [20].....	12
Figura 4.3: Curva Bézier e objeto representado por NURBS [23].....	13
Figura 4.4: Objetos 3D representados por polígonos	14
Figura 4.5: Exemplos de sistemas de partículas.....	15
Figura 4.6: Geometria de um objeto 3D baseado em polígonos.....	16
Figura 4.7: Representação simbólica do pivô de um objeto.....	16
Figura 4.8: Exemplo de textura intercalável.....	17
Figura 4.9: Textura aplicada a uma geometria.....	17
Figura 4.10: Exibição de uma textura em uma geometria mapeada.....	18
Figura 4.11: Utilização de texturas com canal alpha.....	19
Figura 4.12: Diferentes materiais aplicados a uma mesma geometria.....	19
Figura 4.13: Relações entre geometrias, materiais e texturas.....	20
Figura 4.14: Geometria com múltiplas texturas.....	20
Figura 4.15: Representação de um objeto com múltiplas texturas	21
Figura 4.16: Objeto com múltiplos materiais.....	21
Figura 4.17: Material com duas texturas e uma máscara.....	22
Figura 4.18: Funcionamento de uma máscara.....	22
Figura 4.19: Material compartilhado por diferentes geometrias.....	23
Figura 4.20: Textura compartilhada por diferentes materiais.....	23
Figura 4.21: Compartilhamento de texturas e materiais.....	24
Figura 4.22: Objeto com animação skeleton.....	25
Figura 4.23: Interfaces gráficas do 3dsmax e Blender.....	26
Figura 4.24: Interfaces gráficas do Photoshop e Gimp.....	27

Figura 5.1: Interface geral do sistema ITV.....	28
Figura 5.2: Diagrama funcional do sistema ITV.....	30
Figura 5.3: Diagrama de utilização das estruturas XML.....	32
Figura 5.4: Tecnologias envolvidas no sistema ITV e suas áreas de influência.....	33
Figura 5.5: Visão geral da represa da UHE Tucuruí.....	37
Figura 5.6: Desenho 2D com a visão geral da usina.....	39
Figura 5.7: Desenho 2D da planta baixa de uma das cotas.....	40
Figura 5.8: Relação entre os desenhos 2D de referência.....	41
Figura 5.9: Foto de referência e geometrias do terreno.....	43
Figura 5.10: Imagens do terreno no ambiente virtual do sistema ITV.....	44
Figura 5.11: Imagens da água no ambiente virtual do sistema ITV.....	45
Figura 5.12: Objetos de concreto do sistema ITV.....	46
Figura 5.13: Rotor do gerador no ambiente virtual.....	47
Figura 5.14: Tubulações com setas indicativas de fluxo.....	48
Figura 5.15: Grades da tomada d'água (montante).....	49
Figura 5.16: Geometria e texturas da grade do teto.....	49
Figura 5.17: Fotografia do salão de máquinas e sua representação virtual.....	50
Figura 5.18: Grade no piso do salão de máquinas.....	50
Figura 5.19: Textura dos manômetros.....	51
Figura 5.20: Monitores de vídeo da sala de comando.....	52
Figura 5.21: Iluminação do corredor por meio de texturas.....	53
Figura 5.22: Vaso de planta no sala de comando.....	53
Figura 5.23: Textura de rosca de parafusos.....	54
Figura 5.24: Tampa superior do sistema distribuidor (furos inferiores).....	54
Figura 5.25: Tampa superior do sistema distribuidor (furos superiores).....	55
Figura 5.26: Rotor da turbina.....	55
Figura 5.27: Escotilha de acesso à caixa espiral.....	56
Figura 5.28: Mola da válvula detectora de pressão	57
Figura 5.29: Mola do sistema de frenagem.....	57

Capítulo 1 – Introdução

O avanço tecnológico ocorrido na última década, principalmente o referente à eletrônica e à computação, permitiu o surgimento e disseminação de técnicas sofisticadas de interação usuário-máquina. A utilização do computador e outros equipamentos eletrônicos passou por significativas transformações. As telas com texto monocromático deram lugar a ambientes gráficos multimídia, coloridos, com efeitos de sombras e animações. Os ambientes gráficos de interação (interfaces gráficas) com o usuário diminuíram a complexidade de uso de sistemas computacionais, facilitando o aprendizado e controle desses sistemas; além de contribuir para a popularização dos computadores.

A evolução das interfaces e outros recursos computacionais permitiu o uso do computador para os mais variados fins; de aplicações de escritório a sistemas de controle industrial; para simples entretenimento ou para finalidades médicas. Nesse contexto, ambientes virtuais tridimensionais (doravante chamados de ambientes 3D) se configuram como uma tecnologia muito utilizada e em ascensão, presente em muitas aplicações de engenharia, arquitetura e jogos eletrônicos.

Em diversas áreas, incluindo a industrial, as tecnologias 3D geralmente estão presentes em sistemas de treinamento e simulação, cujos resultados tem sido considerados relevantes e satisfatórios [1,2,3], dados os benefícios que ambientes virtuais podem trazer ao aprendizado[4,5] e a possibilidade cada vez maior de representar ambientes reais com o nível de detalhes requerido.

O sistema computacional chamado de ITV (Instruções Técnicas Virtuais), que serve como estudo de caso deste trabalho, é um exemplo de aplicação 3D voltada ao treinamento e simulação industrial. É um ambiente de criação e execução de animações virtuais 3D interativas, que representam instruções de manutenção (montagem/desmontagem) e operação de maquinário da Usina Hidrelétrica de Tucuruí. Este trabalho faz uma abordagem sobre a metodologia utilizada na implementação do sistema ITV, com ênfase à criação dos objetos 3D, considerando requisitos, objetivos e limitações do processo.

O capítulo 2 comenta os resultados obtidos por trabalhos correlatos a este, cujo objetivo também foi treinamentos baseados em ambientes virtuais.

O capítulo 3 comenta alguns métodos de interação do usuário com sistemas computacionais, com destaque à utilização de elementos tridimensionais.

O capítulo 4 aborda métodos de representação de objetos 3D, tipos de aplicação às quais eles se destinam, técnicas de criação de objetos realísticos e como eles se inserem no contexto de um ambiente virtual.

O capítulo 5 situa o sistema ITV no contexto de aplicações 3D, detalhando suas características funcionais, sua arquitetura computacional, a metodologia de organização de seus recursos e mostra alguns exemplos de seus objetos 3D.

O capítulo 6 tece comentários finais e cita possíveis melhorias e novas funcionalidades ao sistema ITV.

Capítulo 2 – Trabalhos Correlatos

As recentes tecnologias computacionais permitiram que novas mídias pudessem ser usadas para os mais variados fins, aproveitando as vantagens que cada recurso pode proporcionar. Nos últimos anos, ambientes tridimensionais interativos têm sido empregados em variados tipos de aplicação, entre eles os voltados a simulações e treinamentos virtuais, como um recurso complementar aos manuais impressos e vídeos. Este capítulo aborda dois trabalhos correlatos ao Sistema de Instruções Técnicas Virtuais, que também utilizaram ambientes virtuais *desktop* (utilizando computadores domésticos, de uso geral) para treinamento e/ou simulação.

2.1 Unidade Geradora Virtual

O sistema intitulado por *Unidade Geradora Virtual (UGV)*, abordado em [1], consiste em um ambiente tridimensional interativo destinado à visualização de determinados procedimentos de manutenção e operação da Usina Hidrelétrica de Tucuruí (que também serve como objeto de estudo do presente trabalho).

Assim como o sistema ITV, a UGV foi projetada para atuar como um recurso facilitador à obtenção dos conhecimentos que antes eram alcançados por treinamentos convencionais baseados em roteiros em papel e, em alguns casos, vídeos. Por outro lado, enquanto o sistema ITV permite ao usuário criar e alterar uma instrução virtual (por meio de um módulo de autoria), a UGV foi implementada de modo que as instruções virtuais formassem um conjunto pré-determinado e imutável; ou seja, como informações *somente-leitura* por parte do usuário-alvo. Estruturalmente, a UGV possui 5 módulos funcionais: *Educativo*, *Operação*, *Montagem*, *Manutenção* e *Maquete Virtual*.

O módulo *Educativo* tem como objetivo a familiarização do usuário com a estrutura geral da usina, permitindo a navegação pelo ambiente e visualização de informações textuais sobre as peças.

O módulo de *Operação* permite a simulação de alguns procedimentos de operação da usina, possibilitando um acompanhamento gráfico de variáveis das equações dinâmicas que modelam o processo [6] e os seus efeitos visuais (rotação da turbina e ângulos de abertura de engrenagens, por exemplo) no ambiente 3D.

O módulo de *Montagem* corresponde ao procedimento de construção e montagem geral de uma das unidades da usina, incluindo a sua concretagem. O módulo de *Manutenção*, por sua vez, compreende os procedimentos de treinamento de montagem e desmontagem de alguns conjuntos mecânicos. A *Maquete Virtual* permite uma visão panorâmica das instalações da usina, incluindo a represa e o rio Tocantins, por meio de uma animação que percorre o ambiente 3D. A figura 2.1 mostra alguns cenários do sistema UGV.

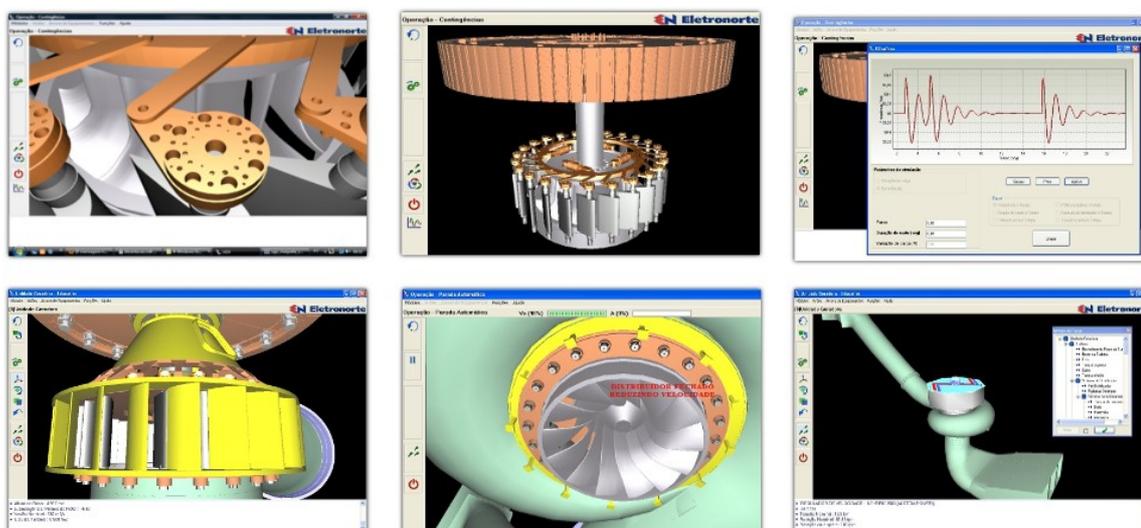


Figura 2.1: Alguns cenários do sistema UGV

O trabalho [1] também aborda alguns dados estatísticos obtidos de um teste de usabilidade baseado em questionários, realizado com 28 alunos de Engenharia Elétrica, que avaliaram, entre outros aspectos, as suas experiências após o treinamento virtual. Uma das perguntas do questionário abordava o aspecto realístico das peças e equipamentos virtuais dos treinamentos de montagem e manutenção. As alternativas referentes a esse item permitiam ao participante classificar o grau de realismo das peças como:

- a) *insatisfatório*;
- b) *pouco satisfatório, sem o reconhecimento de boa parte das peças virtuais*;
- c) *satisfatório, mesmo algumas peças apresentando limitações realísticas*; ou
- d) *bastante satisfatório*.

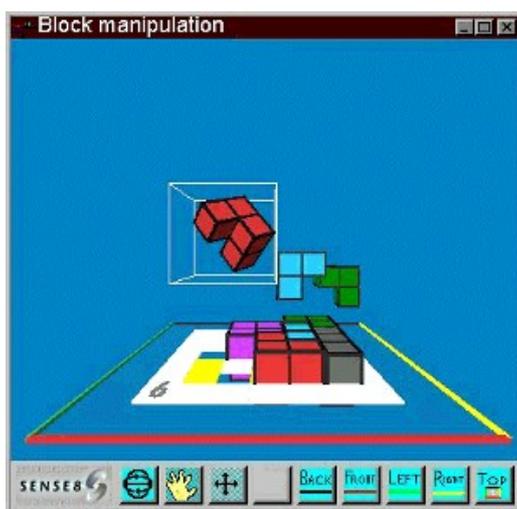
Com relação a esse item, 89% dos participantes avaliaram o grau de realismo das peças como *bastante satisfatório*; enquanto 11% avaliaram que o aspecto visual era *satisfatório, mesmo algumas peças apresentando limitações realísticas*;

2.2 Comparação entre ambientes 3D e outras mídias de treinamento

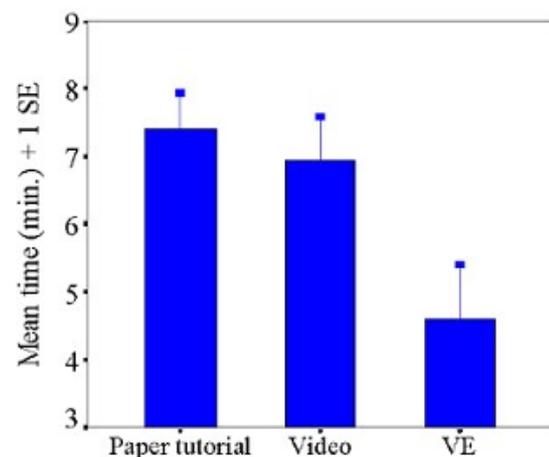
O trabalho [3], baseado em um experimento envolvendo 40 participantes, compara a eficácia de um treinamento por meio de um ambiente virtual *desktop* com relação a dois outros meios: instruções impressas e tutoriais em vídeo.

Os participantes foram instruídos sobre os procedimentos de encaixe de blocos reais com diferentes formas geométricas, utilizando uma das três mídias de treinamento (instruções em papel, vídeo ou ambiente virtual). Os participantes submetidos ao tutorial em vídeo tiveram um tempo de treinamento constante. Os dois outros grupos (utilizando instruções em papel ou ambiente virtual) tiveram acesso à mídia de treinamento durante o tempo que cada um necessitou para completar a tarefa de montagem.

Após o treinamento, os participantes realizaram dois procedimentos de montagem real: o primeiro logo após ao treinamento e o segundo após uma semana. Os tempos de montagem logo após o treinamento não apresentaram significativas diferenças entre os grupos. Por outro lado, após uma semana, o grupo que foi treinado pelo ambiente virtual precisou, em média, de um tempo significativamente menor para completar a montagem. A figura 2.1 mostra a interface do ambiente virtual (a) e um gráfico com as médias (em minutos) dos tempos de montagem para cada grupo: instruções em papel (*paper tutorial*), vídeo e ambiente virtual (VE).



(a) interface gráfica [3]



(b) gráfico comparativo [3]

Figura 2.3: Ambiente 3D para instruções de montagem e gráfico comparativo com outras mídias

Capítulo 3 – Ambientes Virtuais 3D

Sistemas computacionais que possuem interação com o usuário geralmente precisam expor informações visuais, frequentemente exibidas por dispositivos como monitores ou projetores de vídeo. O ambiente gráfico que o usuário visualiza (e com o qual interage) é comumente chamado de *GUI*, da expressão *graphical user interface* em inglês, que corresponde a *interface gráfica do usuário*, em português. Interfaces que representam espaços tridimensionais constituem os *ambientes virtuais 3D*, que possuem mecanismos de interação e implementação diferenciados com relação a ambientes GUI 2D.

3.1 Interação e Imersão

A interação do usuário com ambientes 3D pode ocorrer por diversas formas. Um ambiente 3D pode ser classificado, dentre outros critérios, pelo método como o usuário interage e recebe informações do sistema. Ambientes classificados como *fisicamente imersivos* utilizam dispositivos especiais para esse tipo de interação, como luvas com sensores que captam os movimentos do usuário e, em alguns casos, impõem resistência (a esses movimentos) para fornecer sensações de força, pressão, peso, etc [7]. A figura 3.5 mostra exemplos de dispositivos usados para interação com ambientes 3D. A figura 3.5(a) mostra equipamentos criados na década de 1980. Em 3.5(b) é mostrada a *CyberGlove II*, uma luva com transmissão de dados sem fio.



(a) Dispositivos da década de 1980 [8]



(b) Luva CyberGlove II [9]

Figura 3.1: Luvas para interação com ambientes virtuais

Alguns jogos eletrônicos que simulam corridas automobilísticas permitem interação por meio de controladores que imitam o volante, pedais e alavanca de câmbio de veículos reais. Alguns modelos desses controladores possuem engrenagens capazes de girar o volante de acordo com a situação do carro virtual. Assim, o jogador pode, por exemplo, perceber que precisa segurar o volante de forma mais rígida para estabilizar as rodas em caso de problema com um dos pneus. A figura 3.6 mostra um jogador (a) utilizando o volante *MOMO Racing Force Feedback* (b), fabricado pela Logitech [10].



(a) Jogador utilizando um volante



(b) Controlador *Logitech MOMO Racing Force Feedback*

Figura 3.2: Controlador de jogos de corrida

Além da imersão física, ambientes virtuais também podem ser qualificados como *mentalmente imersivos*, quando o usuário tem uma profunda sensação de participação e envolvimento no sistema, mesmo sem estímulos corporais [7].

Ambientes interativos implementados em computadores pessoais *desktop* (de uso doméstico), utilizando dispositivos de uso geral (como *mouse* e teclado), podem ser classificados como sistemas com imersão mental, uma vez que ocorra uma forte sensação de envolvimento e interação por parte do usuário. Muitas vezes, esses sistemas utilizam interfaces GUI em conjunto com os cenários 3D.

3.2 Interfaces

O sistema ITV, abordado neste trabalho, é um exemplo de ambiente 3D altamente interativo, que utiliza uma interface GUI auxiliar. Entre outros sistemas com essas características, pode-se citar o *Google Earth* [11], que permite navegação por um ambiente com edificações 3D e referências geográficas.

A figura 3.7 mostra a interface geral (a) do Google Earth e uma visão em detalhe (b) de elementos GUI no interior do ambiente 3D.



Figura 3.3: Interface e ambiente 3D do software Google Earth

A figura 3.8 mostra um ambiente criado com a ferramenta *Unit 3D* [12], destinada ao desenvolvimento de jogos. O cenário representa o compartimento de uma casa, com a possibilidade de movimentar a mobília e controlar a iluminação por meio do *mouse*. O ambiente possui componentes GUI, mostrados na figura 3.8(b), que permitem selecionar quais luzes devem estar ativas.



Figura 3.4: Ambiente 3D com controle de iluminação por meio de elementos GUI [13]

Capítulo 4 – Objetos 3D

Os objetos 3D são elementos essenciais de um cenário tridimensional, uma vez que representam as entidades materiais do mundo real emulado pelo ambiente virtual. De maneira geral, objetos e cenários 3D podem ter diversas finalidades, como produção de filmes, jogos, manuais técnicos, simulações, *etc.* De acordo com o uso ao qual o objeto se destina, diferentes técnicas de criação, representação e realismo podem ser adotados.

4.1 Aplicações-alvo

Considerando as várias finalidades dadas a um objeto 3D, pode-se diferenciar as aplicações, entre outros critérios, em relação à necessidade de interação com o observador, ao nível de detalhes requerido, ou à liberdade de visualização.

A indústria cinematográfica é um exemplo de utilização de cenários 3D não-interativos. O espectador de um filme é, a rigor da palavra, um mero observador, que não pode alterar o cenário virtual. De maneira geral, um cenário 3D não-interativo dispensa um processamento dinâmico; ou seja, os cálculos necessários à obtenção do cenário não precisam ser realizados continuamente, conforme a demanda. Como conseqüência, ambientes 3D sem interação são menos prejudicados por limitações de *hardware*, porque não exigem respostas em tempo-real. O processamento para obtenção de um filme, por exemplo, pode consumir um tempo significativamente grande; porém, uma vez processado, novos cálculos não são necessários. Objetos 3D para filmes podem, portanto, ser mais complexos e detalhados que aqueles para sistemas de tempo-real.

Ambientes interativos que oferecem um elevado grau de liberdade de movimentos (manipulação de câmera e/ou objetos) ao o usuário, exigem estruturas 3D mais elaboradas do que aqueles com restrições de manipulação. Em muitos jogos, o jogador não possui controle total sob a câmera. Entre os jogos de corrida, por exemplo, a parte inferior do veículo não pode ser visualizada na maioria dos casos. Esse comportamento permite a criação de objetos baseados apenas na aparência da estrutura externa, o que caracteriza os objetos conhecidos pela palavra inglesa *boundary* [14], que, nesse contexto, pode ser entendida como “baseado nas bordas”.

Em contrapartida, um objeto 3D destinado à impressão de um manual técnico, para representar uma peça industrial por completo, precisa ser construído levando em consideração também o interior, com fidelidade aos encaixes e ressaltos que o objeto real possui e, possivelmente, com atributos físicos (massa, centro de gravidade, etc). Esses objetos, por sua vez, são considerados *sólidos*, em contraste aos *boundary*.

Em jogos eletrônicos, a utilização de objetos simplificados é possivelmente tolerada pelos jogadores, uma vez que o principal objetivo é o entretenimento. Muitos jogos permitem até que o usuário escolha o nível de detalhes do ambiente, conforme a capacidade de processamento de seu equipamento. Por outro lado, um sistema para estudo de anatomia humana, com visualização tridimensional de ossos e órgãos, provavelmente será mais rigoroso com relação ao nível de detalhes.

A figura 4.1 mostra exemplos de objetos 3D usados para diferentes aplicações. Em 4.1(a) pode-se observar uma cena do jogo *Grand Theft Auto – San Andreas* (GTA) [15], que por exigir processamento em tempo-real (dada a interatividade com o usuário), utiliza objetos simplificados por motivos de desempenho. Na imagem é possível observar que o tronco da árvore é formado por poucos segmentos.

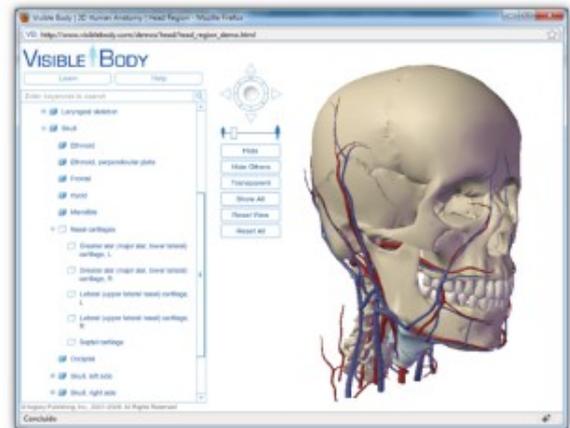
Na figura 4.1(b) é mostrada a aplicação *Visible Body* [16] (destinada ao estudo de anatomia humana), que também é um ambiente interativo, cuja câmera pode ser rotacionada em qualquer um dos três eixos, permitindo a visualização de objetos em qualquer ângulo. Nesse caso, o nível de detalhes dos objetos é um parâmetro fundamental à eficácia da aplicação, uma vez que objetos simplificados poderiam comprometer o entendimento.

Nas imagens em 4.1(c) é mostrado o personagem *Sullivan* do filme *Monsters Inc.* (*Monstros S/A*, no Brasil), que possui mais de 2 milhões de pêlos [17] espalhados pelo corpo. Durante as animações, esses pêlos sofrem influências do movimento do personagem e de agentes externos, simulando a dinâmica de pêlos reais. Tal tarefa poderia ser impraticável para um ambiente interativo; mas, por se tratar de um filme, é factível mesmo que haja demora no processamento.

Na figura 4.1(d) são mostradas algumas peças de um motor de combustão interna do tipo *Wankel*, destinadas à documentação técnica visual. Nesses objetos, pode-se observar que a ênfase foi dada aos detalhes estruturais das peças reais, e não às cores.



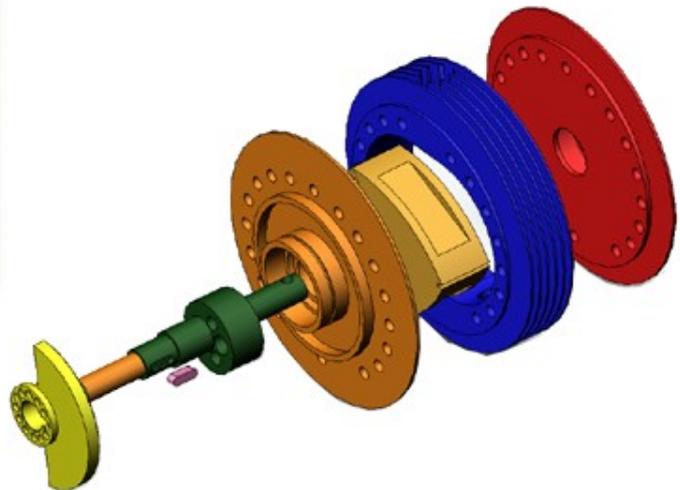
(a) Tronco de uma árvore no jogo GTA



(b) Crânio humano na aplicação Visible Body



(c) Personagem do filme "Monstros S/A" [18]



(d) Peças de um motor *wankel* [19]

Figura 4.1: Diferentes níveis de detalhamento de objetos 3D de acordo com a finalidade

4.2 Métodos de Representação Geométrica

A estrutura geométrica tridimensional que forma um objeto pode ser obtida por diferentes métodos. Dentre as técnicas de representação, podem ser citadas as baseadas em *voxels*, em curvas parametrizadas e em polígonos.

4.2.1 Voxels

O termo *voxel* é usado para expressar um pixel volumétrico; ou seja, um pixel situado num espaço tridimensional. Objetos 3D podem ser representados por uma matriz tridimensional de *voxels*, na qual elementos vazios representam pontos não preenchidos pelo objeto, como mostra a figura 4.2.

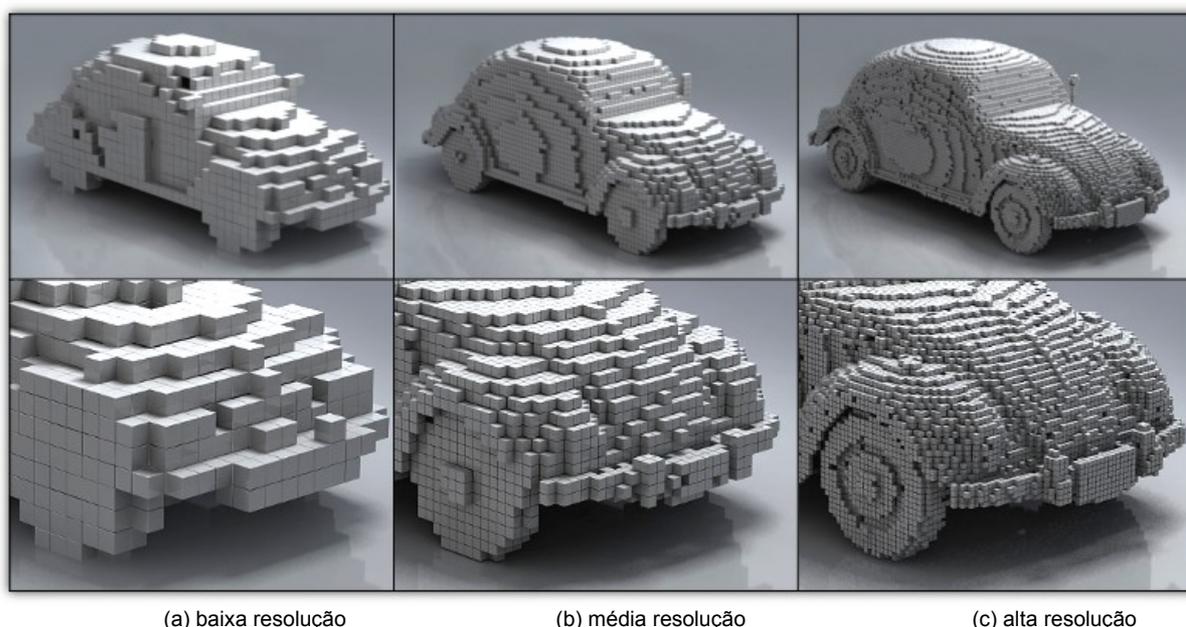


Figura 4.2: Aspecto visual de um objeto formado por *voxels* [20]

Comparando com outras técnicas, objetos baseados em *voxels* demandam uma grande consumo de memória para altas resoluções [13] e, por isso, são indicados para representar apenas superfícies muito irregulares ou quando um elevado nível de precisão é requerido. Algumas aplicações médicas, como exames radiológicos, utilizam imagens baseadas nesse princípio [21], para representar a profundidade de imagens em cortes seccionais. Uma matriz de *voxels* constitui o resultado primário de um processo de escaneamento (*scanning*) 3D, que geralmente é convertido, posteriormente, para outro método de representação.

4.2.2 Curvas Parametrizadas

Curvas parametrizadas, utilizadas em estruturas vetoriais 2D e 3D, constituem um meio de representação baseado em equações e parâmetros, que permitem a determinação numérica dos pontos pertencentes a uma curva ou superfície [22]. Entre as técnicas mais usuais, podem ser citadas as *curvas de Bézier* e as *NURBS (Non-Uniform Rational Basis Splines)*. A figura 4.3(a) mostra uma curva Bézier (em vermelho) e seus *pontos de controle*, que servem de parâmetros. Em 4.3(b) é mostrado um barco 3D construído com curvas NURBS.

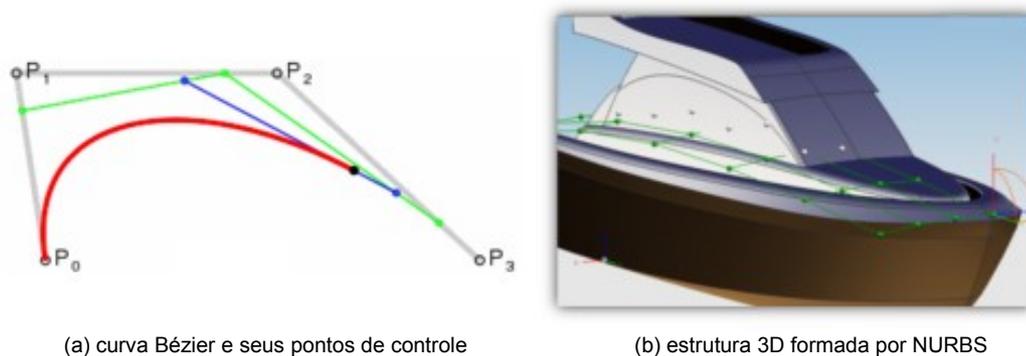


Figura 4.3: Curva Bézier e objeto representado por NURBS [23]

Em geral, a utilização de curvas parametrizadas facilita a criação de superfícies relativamente complexas, por meio da manipulação interativa dos pontos de controle. Por outro lado, objetos 3D criados com NURBS, por exemplo, geralmente são convertidos para uma representação baseada em polígonos em um dado momento de sua utilização ou mesmo durante sua renderização [22,24,25].

4.2.3 Polígonos

Assim como as curvas parametrizadas, as estruturas baseadas em polígonos também são vetoriais, mas implementam curvas por aproximação. Em comparação a outros métodos, objetos formados por polígonos são de fácil manipulação computacional [26] e por esse motivo são amplamente utilizados. De fato, até os objetos definidos por curvas parametrizadas são geralmente convertidos para superfícies segmentadas em polígonos [22] (processo chamado de *tessellation* [24]). Com relação à geometria, uma estrutura 3D baseada em polígonos é um poliedro (ou um conjunto de polígonos), mesmo que o objetivo seja representar uma esfera,

por exemplo. Essa estrutura geométrica é comumente chamada de *mesh*, fazendo referência à malha (*mesh*, em inglês) de vértices e arestas que formam o poliedro.

A grande maioria das técnicas de representação poligonal utiliza triângulos para formar esses poliedros [24]; ou seja, objetos 3D baseados em polígonos são implementados como poliedros de faces triangulares, mesmo que o objeto represente, visualmente, um cubo. A figura 4.4(a) mostra as faces convencionais (segundo os conceitos da geometria espacial) de um paralelepípedo. Na figura 4.4(b) são mostradas as faces efetivas; ou seja, a estrutura que é implementada computacionalmente. Em 4.4(c) é mostrada uma esfera formada por triângulos. Em 4.4(d) é exemplificada uma representação poligonal de um golfinho.

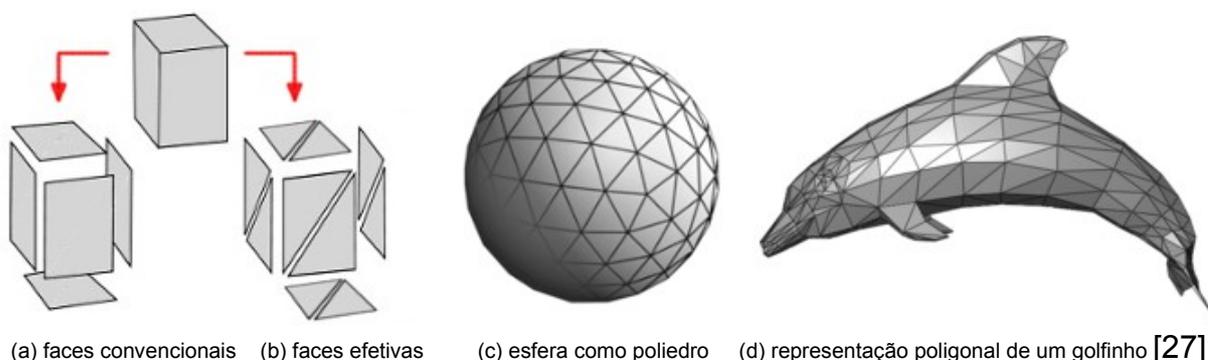


Figura 4.4: Objetos 3D representados por polígonos

4.2.4 Sistemas de Partículas

Sistema de partículas é uma técnica especialmente utilizada para representar fenômenos de comportamento aleatório que não possuem uma estrutura geométrica constante, tais como fumaça, fogo, explosões, faíscas, chuva, neve, *etc.* Técnicas semelhantes também podem implementar outros eventos não-determinísticos, como o movimento de fios de cabelo ou pêlos, exemplificado na figura 4.1(c), que mostra um personagem do filme *Monstros S/A*.

A representação computacional de partículas geralmente é feita por regras e parâmetros de comportamento, como quantidade de partículas, tempo médio de vida de uma partícula, pontos emissores, frequência de emissão, direção predominante, influência de forças externas, *etc* [28]. Em geral, sistemas de partículas exigem um processamento relativamente elevado, um vez que demandam numerosos cálculos para determinar o movimento de cada partícula.

Por serem desprovidos de uma forma geométrica sólida e constante, sistemas de partículas possuem um processo de criação peculiar, significativamente diferente do processo de construção de objetos baseados em polígonos ou curvas parametrizadas. A figura 4.5 mostra dois exemplos de sistemas de partículas, criados pelo software *Particle Accelerator* [29].

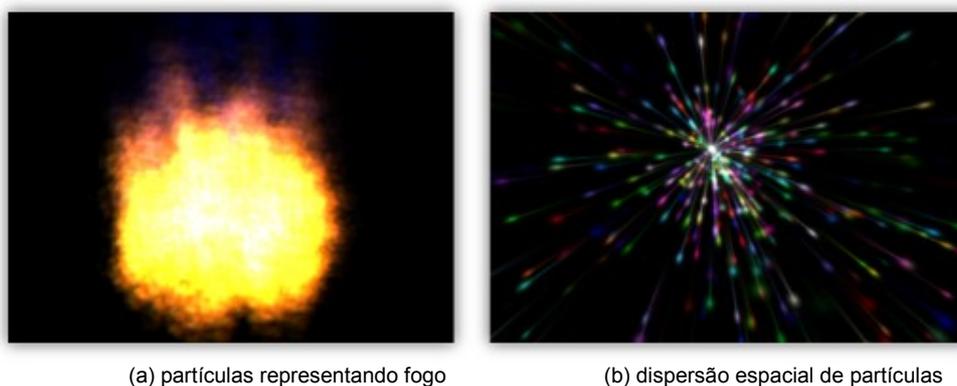


Figura 4.5: Exemplos de sistemas de partículas

4.3 Objetos realísticos e otimizados

Muitas aplicações 3D demandam objetos e cenários tão realísticos quanto possível. Entre as técnicas mais utilizadas para esse fim, está a representação de objetos por polígonos texturizados [30], que por serem de fácil manipulação computacional [26], são indicados para ambientes 3D que exigem resposta em tempo-real. Esse tipo de objeto é frequentemente estruturado por três principais partes: *geometria*, *textura* e *material*.

4.3.1 Geometria

A *geometria* é a parte vetorial de um objeto 3D, que constitui a estrutura geométrica espacial que o representa. Em outras palavras, a geometria pode ser entendida como uma entidade concreta, que exerce o papel de alicerce/armação do objeto. A figura 4.6 mostra a geometria de um objeto, baseado em polígonos, que representa uma ferramenta conhecida como *chave inglesa*. Na figura 4.6(a) são mostrados os polígonos que compõem o poliedro; e em 4.6(b) uma visualização em *wireframe*, que mostra apenas as arestas.

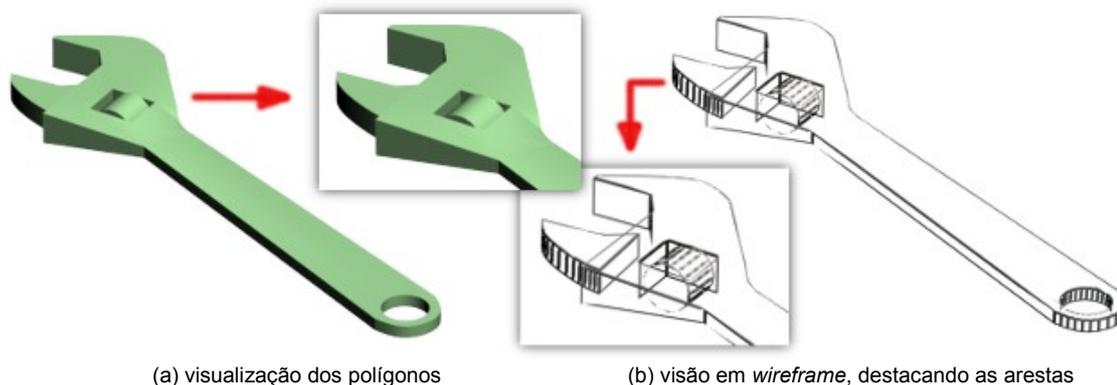


Figura 4.6: Geometria de um objeto 3D baseado em polígonos

Tipicamente, cada geometria tem um *pivô*, que pode ser entendido como o ponto gravitacional do objeto, servindo de referência para o posicionamento, transformações de escala e rotações no contexto do mundo virtual em que o objeto está inserido. Em termos práticos, o pivô é uma estrutura abstrata, que atua como um sistema de eixos interno, como se o objeto fosse um “sub-universo” e seus polígonos os “sub-objetos”. Desse modo, o pivô pode ser interpretado como o ponto *origem* do objeto. A figura 4.7 ilustra, simbolicamente, os eixos que formam um pivô de um objeto, visto em 3 projeções ortogonais (*wireframe*) e em perspectiva.

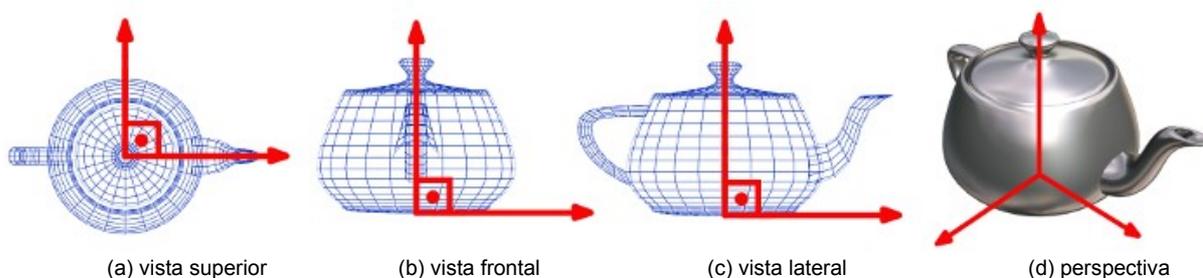


Figura 4.7: Representação simbólica do pivô de um objeto

4.3.2 Textura

As *texturas* são imagens que constituem a parte bitmap de um modelo, contribuindo de forma significativa para obtenção de um aspecto visual realístico[31]. Uma das principais finalidades das texturas é fornecer a percepção da substância da qual o objeto é composto (madeira, concreto, metal, *etc*), representando as características peculiares de sua superfície.

Conceitualmente, as texturas são projetadas para serem *intercaláveis*, de modo a proporcionar um aspecto de continuidade quando replicadas lado a lado. No caso das texturas *não-intercaláveis*, pode-se dizer que o número de repetições é unitário. A figura 4.8 mostra uma textura que representa uma superfície formada por tijolos e a percepção de continuidade após replicações.

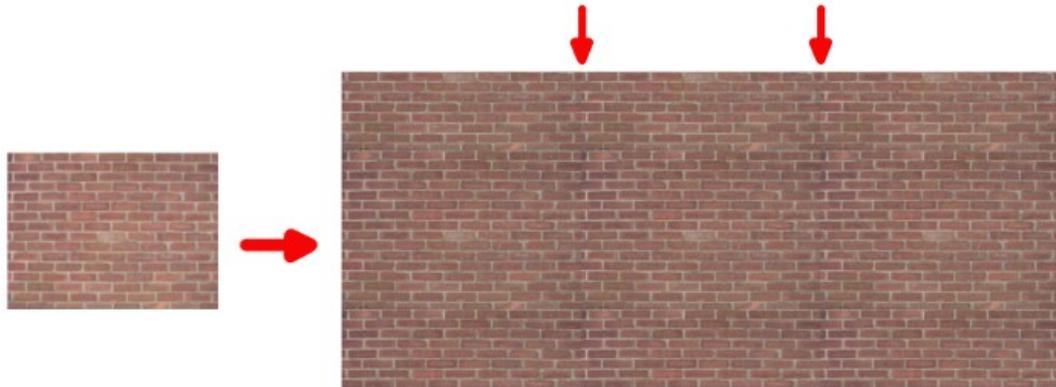


Figura 4.8: Exemplo de textura intercalável

Muitas vezes, as texturas permitem a utilização de geometrias mais simples, já que alguns detalhes estruturais, como pequenos ressaltos nas superfícies, podem ser representados por texturas. A figura 4.9 mostra uma textura não-intercalável obtida a partir de uma fotografia de uma chave inglesa e o resultado visual do objeto 3D após a textura ser aplicada à geometria.

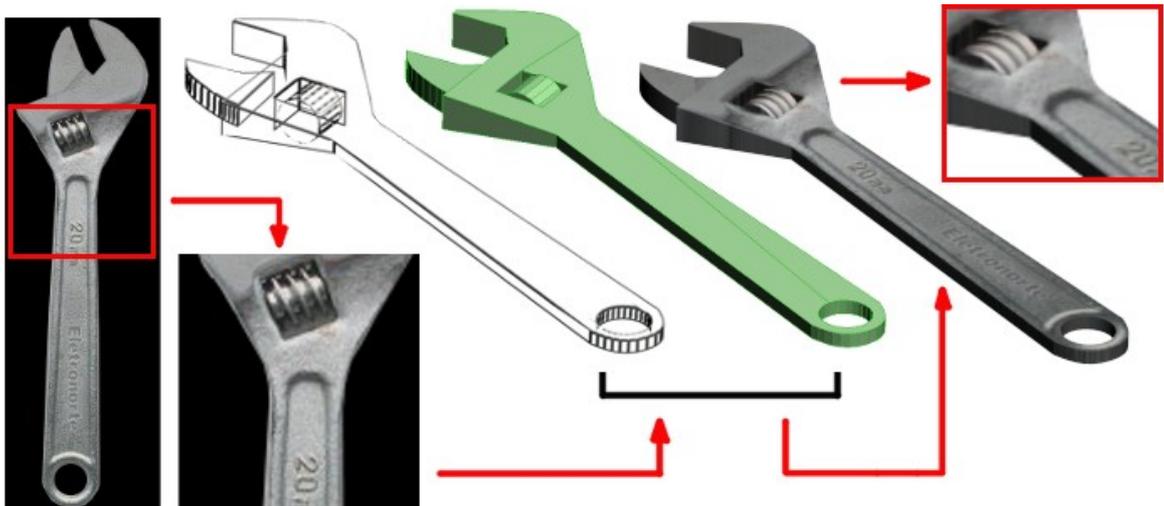


Figura 4.9: Textura aplicada a uma geometria

4.3.2.1 Mapeamento

Para que as texturas, principalmente as não-intercaláveis, sejam devidamente exibidas nas superfícies da geometria, é necessário que sejam determinadas coordenadas para associar a área de cada polígono a uma área da textura [32]. O processo de definição dessas coordenadas é conhecido como *mapeamento*. De forma prática, esse processo consiste em distribuir referências (cópias simbólicas) dos polígonos (que compõem a geometria) em uma superfície bidimensional, para que sejam adequadamente posicionados nas devidas regiões da textura. A figura 4.10 mostra um objeto em que alguns polígonos foram mapeados sob a região azul e outros sob a região com o círculo.

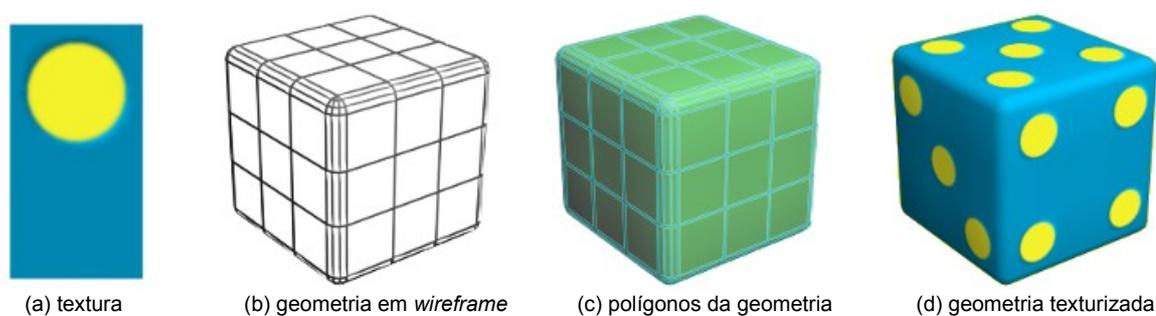


Figura 4.10: Exibição de áreas específicas de uma textura em uma geometria mapeada

Uma vez que o mapa de coordenadas de textura está diretamente associado à geometria do objeto, as informações do mapeamento geralmente são armazenadas juntamente com a própria geometria [32]; ou seja, o mapeamento é uma característica intrínseca à geometria.

4.3.2.2 Transparência

Um recurso adicional que pode ser realizado por meio de texturas é a criação de áreas transparentes na geometria. De fato, a geometria não é alterada; ou seja, continua com o mesmo número de polígonos, mas as regiões especificadas para serem transparentes não são exibidas. Essa prática é muito comum para evitar a criação de geometrias com muitos polígonos.

Em alguns formatos de imagens, como o PNG, já é prevista a utilização de regiões transparentes na própria especificação do formato, estabelecendo uma estrutura lógica conhecida como *canal alpha*. Em outros casos, pode-se definir uma determinada cor da imagem para ser considerada como transparente.

A figura 4.11 mostra a diferença de efeito entre uma textura com canal *alpha* e outra sem, aplicadas a uma mesma geometria.

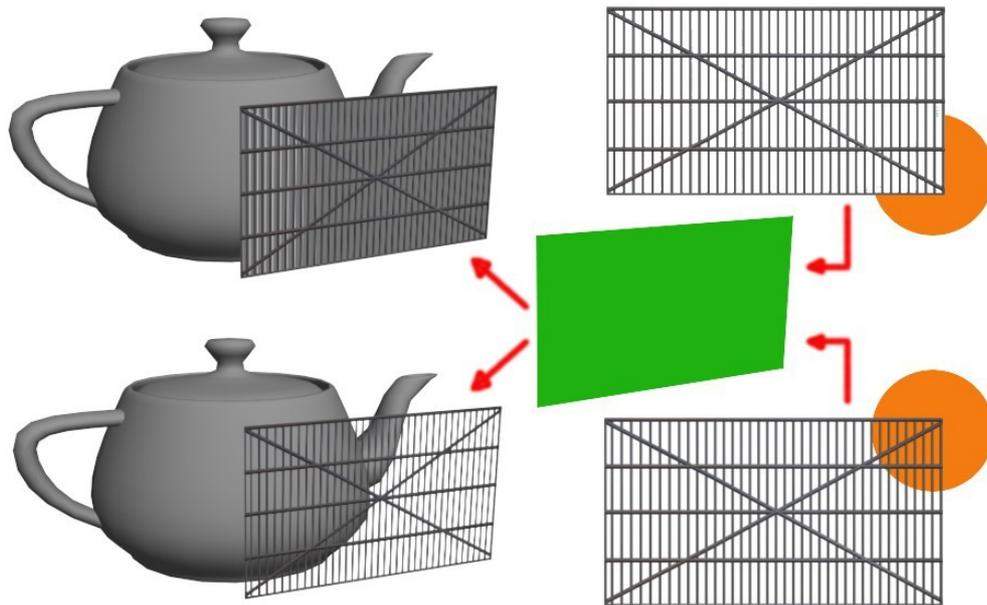


Figura 4.11: Utilização de texturas com canal *alpha*

4.3.3 Material

Um *material* é formado por um conjunto de propriedades e parâmetros que controlam os aspectos visuais dos objetos [33,34], como nível de transparência, brilho, cor, reflexibilidade, refratividade e até mesmo características de exibição das texturas (número de repetições, por exemplo). A figura 4.12 mostra diferentes configurações de material aplicados a uma mesma geometria. Nesse exemplo, não foram utilizadas texturas (utilizou-se apenas uma cor, definida pelo material) para enfatizar as possíveis configurações.

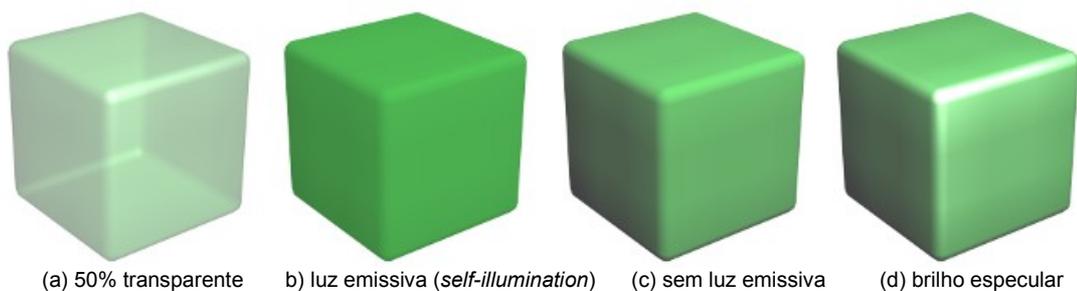


Figura 4.12: Diferentes materiais aplicados a uma mesma geometria

Conceitualmente, o material é uma entidade abstrata que, além de definir características visuais (brilho, transparência, *etc*), também estabelece a ligação entre geometria e textura (entidades concretas). De fato, as texturas são um dos parâmetros do material; ou seja, é o material que associa a textura à geometria. Ou seja, enquanto a *textura* é responsável pela percepção de qual substância compõe a superfície do objeto, é o *material* que parametriza as texturas. Por esses motivos, é comum o emprego dos termos *material* e *textura* como se fossem sinônimos. A figura 4.13 mostra um diagrama que representa, de maneira geral, as relações entre geometrias, materiais e texturas que formam os objetos 3D.

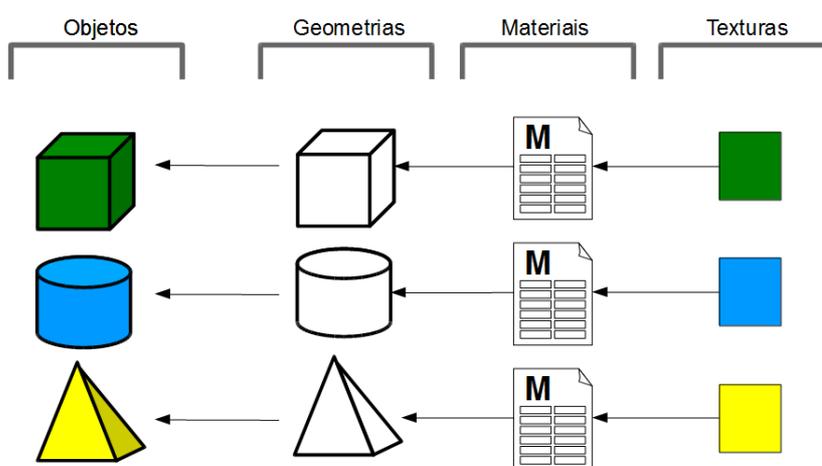


Figura 4.13: Relações entre geometrias, materiais e texturas

4.3.3.1 Sub-materiais

Objetos complexos precisam, muitas vezes, utilizar mais de uma textura em uma mesma geometria. Nesses casos, o material usado pode ser dividido em partes que exercem papéis de *sub-materiais*. A figura 4.14 mostra um objeto com 3 texturas.

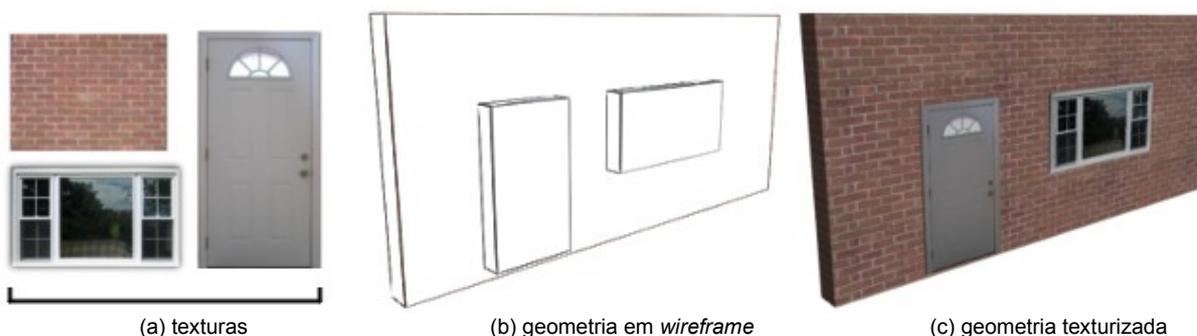


Figura 4.14: Geometria com múltiplas texturas

A figura 4.15 ilustra o efeito de múltiplas texturas em um objeto e suas relações com o material. Como pode ser observado neste exemplo, cada polígono exibe uma textura diferente.

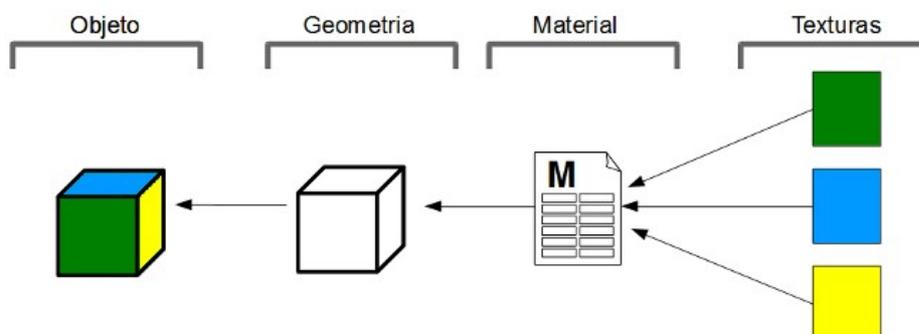


Figura 4.15: Representação de um objeto com múltiplas texturas (sub-materiais)

Dependo da implementação do ambiente virtual, o conceito de sub-material pode não ser aplicado. Nesse caso, pode-se admitir que os sub-materiais são materiais de fato, e que os objetos podem possuir mais de um material. O efeito dessa convenção é meramente organizacional, sem implicações efetivas às relações entre geometria, material e textura. Nesse ponto de vista, a utilização de várias texturas pode ser representada pelo diagrama mostrado na figura 4.16.

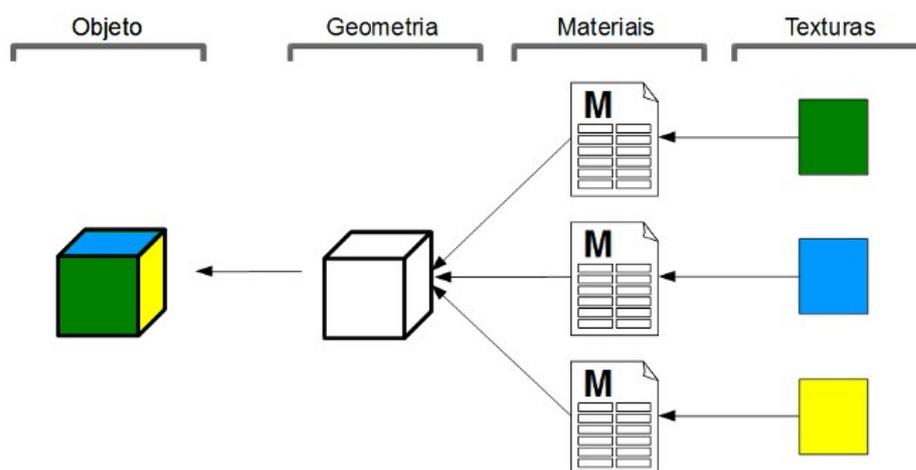


Figura 4.16: Objeto com múltiplos materiais

4.3.3.2 Máscaras

Máscaras atuam com texturas auxiliares, não-visíveis, que determinam a área de exibição das texturas efetivas. É um recurso útil para separar as áreas de exibição de duas (ou mais) texturas projetadas em um mesmo polígono (com o mesmo mapeamento, conseqüentemente), permitindo também implementar áreas transparentes sem utilizar imagens com canal *alpha*. Geralmente são usadas imagens em escala de cinza para atuarem como máscaras, usando as cores (preto e branco) para separar as áreas de exibição. A figura 4.17 mostra um objeto que utiliza as texturas (a) e (b) nos mesmos polígonos, determinando a área de cada uma pela máscara (c). Onde a máscara é preta é exibida a textura de tijolos; onde é branca, a textura de tinta.

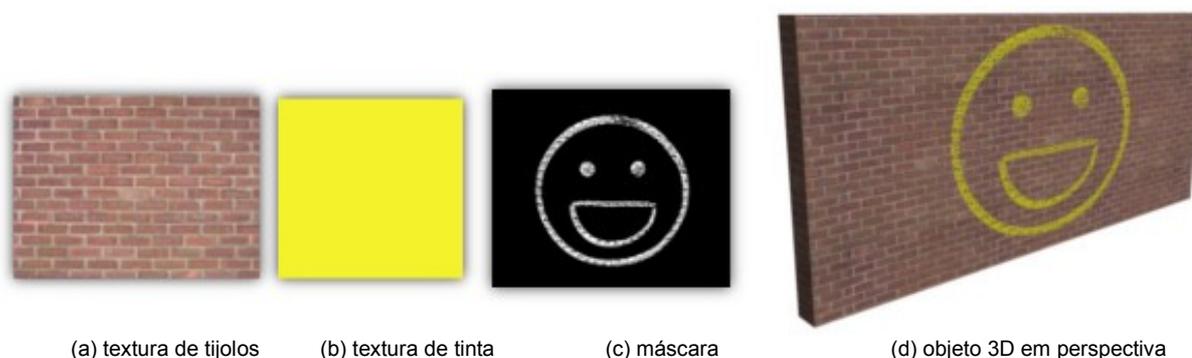


Figura 4.17: Material com duas texturas e uma máscara

A figura 4.18 ilustra um objeto que utilizou uma máscara para separar as áreas de exibição das texturas e como esse elemento se insere no contexto.

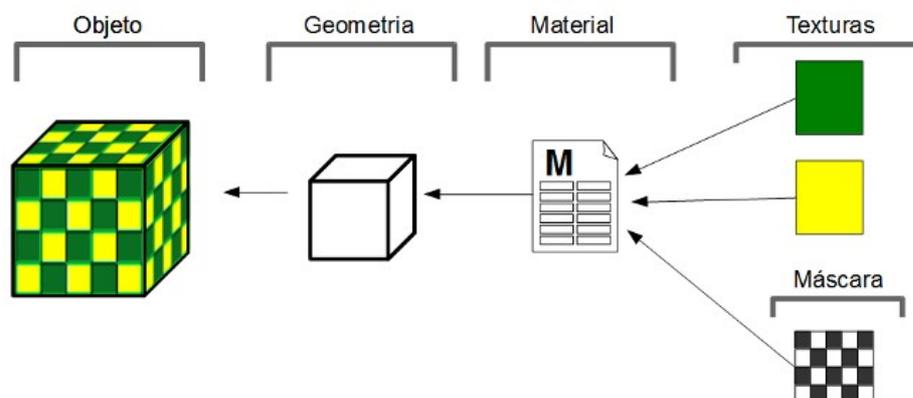


Figura 4.18: Funcionamento de uma máscara

4.3.3.3 Materiais para múltiplos objetos

Em um ambiente típico, um mesmo material pode ser atribuído a objetos diferentes; ou seja, os materiais (e as texturas, conseqüentemente) podem ser simultaneamente compartilhados por mais de um objeto, como ilustra a figura 4.19.

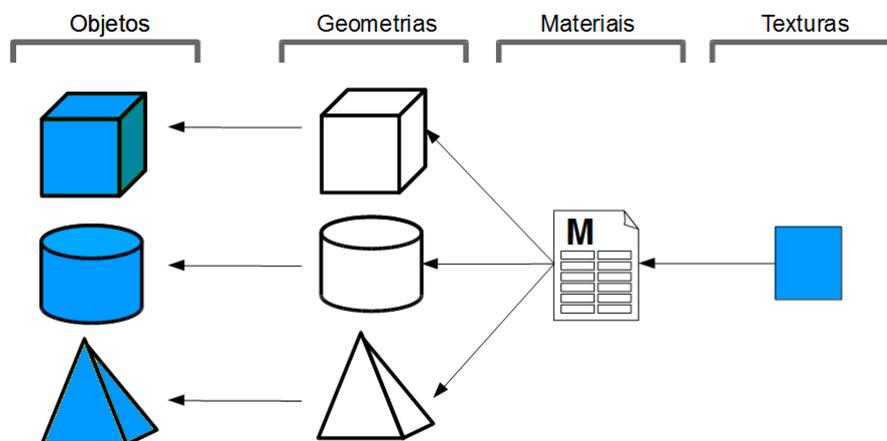


Figura 4.19: Material compartilhado por diferentes geometrias

Outra possível configuração permite que materiais diferentes utilizem a mesma textura, como mostra a figura 4.20. Assim, é possível atribuir diferentes propriedades de exibição (brilho, transparência, etc) a objetos que utilizam a mesma textura.

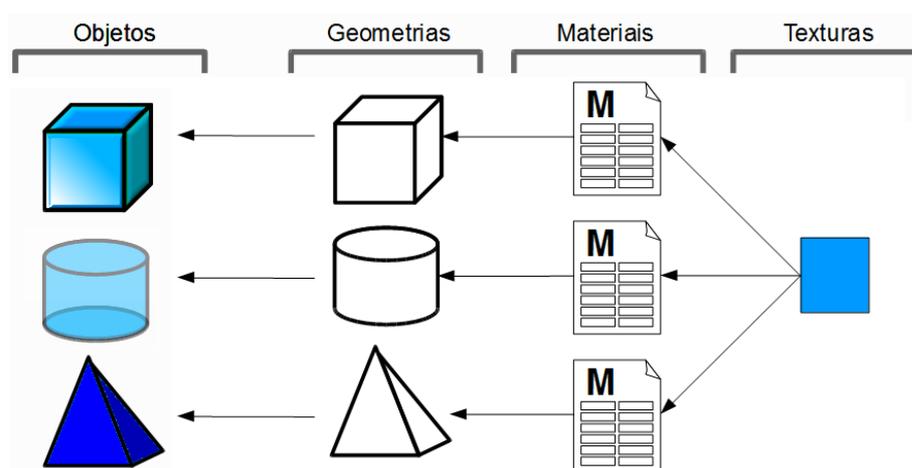


Figura 4.20: Textura compartilhada por diferentes materiais

Nesse contexto, o compartilhamento de materiais e texturas em um ambiente virtual típico pode ser representado pelo diagrama da figura 4.21.

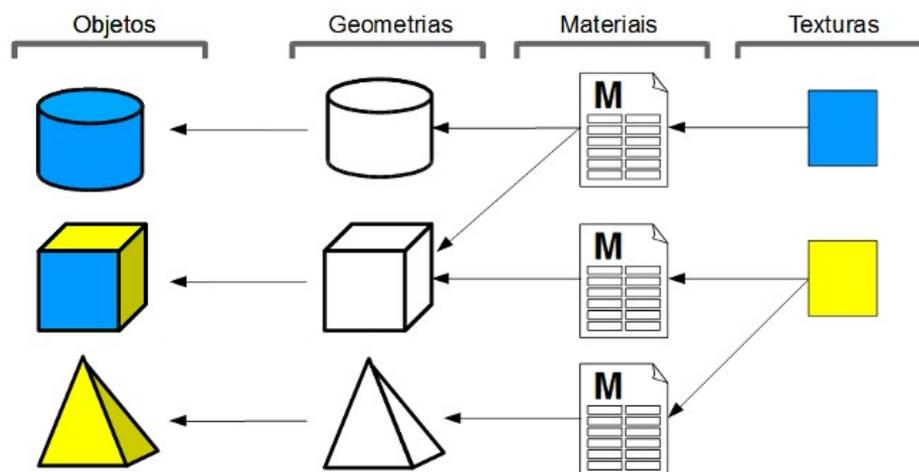


Figura 4.21: Compartilhamento de texturas e materiais

4.4 Animações embutidas

Em um ambiente interativo, animações de translação, de rotação ou de escala são ações freqüentes e facilmente executadas pelo usuário que interage com o ambiente, uma vez que envolvem a manipulação da geometria em sua totalidade. Modificações de escala, por exemplo, compreendem o afastamento ou aproximação de todos os vértices da geometria de forma proporcional. Em rotações, todos os vértices são rotacionados tendo como base o mesmo eixo. Em translações, é necessário apenas somar um mesmo vetor a cada vértice.

Por outro lado, animações mais sofisticadas exigem que subconjuntos de vértices sejam manipulados de forma diferente do restante. Essas animações são geralmente definidas durante o processo de criação do objeto e, por isso, podem ser consideradas como animações embutidas. Assim, antes mesmo de o objeto ser inserido em um ambiente, as animações embutidas já estão definidas em sua estrutura.

Uma técnica muito utilizada para animações embutidas é conhecida como *skeletal animation*, que consiste em utilizar uma estrutura vetorial auxiliar (denominada *skeleton*, ou *esqueleto*) para ser animada e atuar como referência de movimento para os vértices do objeto.

Um *skeleton* é uma estrutura formada por elementos chamados *bones* (ou ossos), não-renderizáveis e frequentemente articulados, que são associados a determinados subconjuntos de vértices da geometria. Durante a animação, cada vértice é movimentado em sincronia com o *bone* a ele associado. Em algumas implementações, é possível associar mais de um *bone* para cada vértice e definir fatores de influência (pesos).

Animações *skeleton* são muito utilizadas para movimentos de personagens e humanóides de maneira geral (também chamados de *avatares*), definindo ações de andar, pular e correr, por exemplo. A figura 4.22 mostra a geometria (já texturizada) de um avatar e seu *skeleton*.

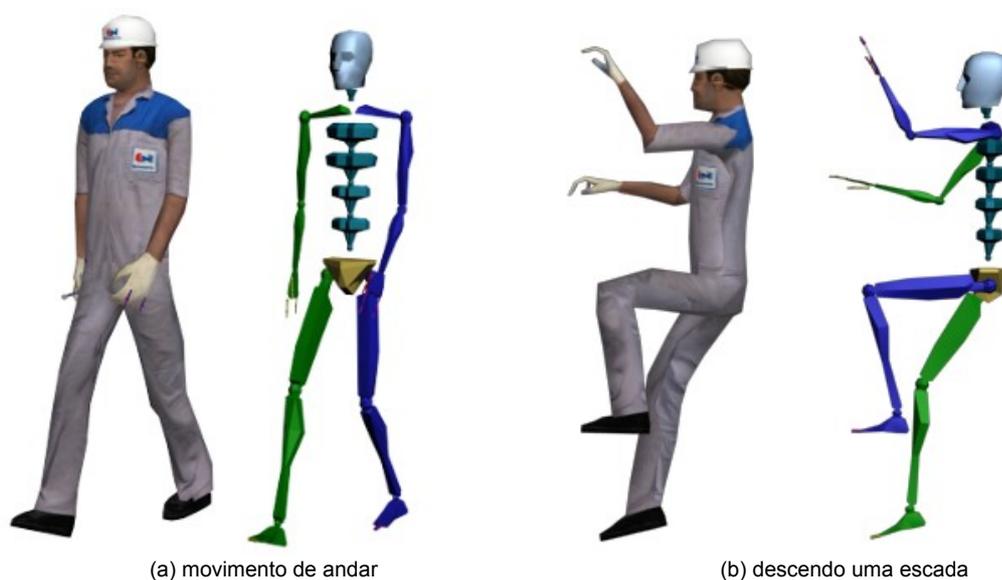


Figura 4.22: Objeto com animação *skeleton*

4.5 Criação

Objetos 3D podem ser criados, entre outras formas, por definição de códigos (escritos diretamente em uma dada linguagem de modelagem), por *scanners* 3D, ou por softwares especializados em tal tarefa. A criação de objetos otimizados baseados em geometria poligonal, textura e material geralmente envolve softwares de desenho 3D e de edição de imagens bitmaps, apresentando muitas semelhanças com processos artísticos e artesanais.

4.5.1 Softwares modeladores 3D

Os softwares de desenho 3D (ou modeladores 3D) participam da etapa de construção da geometria, definição de materiais e criação de animações embutidas (como as baseadas em *skeleton*, por exemplo).

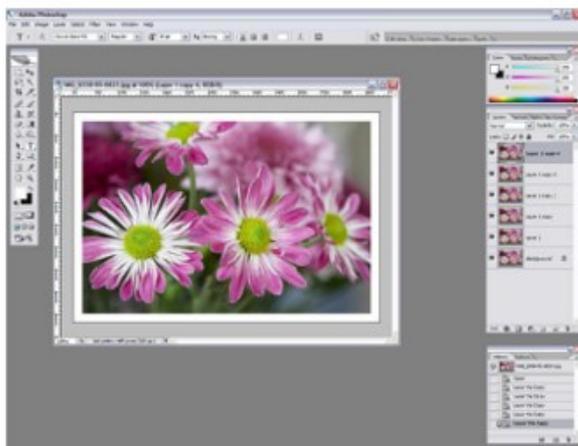
Entre os diversos softwares de modelagem existentes, podem ser citados o *Autodesk 3dsmax* [35] e o *Blender* [36]. O *3dsmax* é um software proprietário, presente no mercado desde a década de 1990, voltado principalmente à criação de objetos 3D para jogos e animações. De forma semelhante, o *Blender* também é principalmente destinado a esses tipos de aplicações, mas ao contrário do *3dsmax*, é um software livre e pode ser executado tanto no ambiente *Windows* quanto em distribuições *Linux*. A figura 4.23 mostra a interface gráfica padrão desses softwares.



Figura 4.23: Interfaces gráficas do 3dsmax e Blender

4.5.2 Softwares de Edição de Imagens Bitmap

Os softwares de edição de imagens bitmaps (geralmente fotografias) atuam no processo de criação das texturas, definindo, por exemplo, se a textura é ou não intercalável. Nessa categoria de software, podem ser citados o *Adobe Photoshop* [37] e o *Gimp* [38], que possuem muitas semelhanças funcionais. Um das principais diferenças entre esses dois softwares é o tipo de licença: enquanto o *Photoshop* é um software proprietário, o *Gimp* é um software livre. A figura 4.24 mostra a interface desses softwares.



(a) Photoshop



(b) Gimp

Figura 4.24: Interfaces gráficas do Photoshop e Gimp

Capítulo 5 – Estudo de caso: Sistema de Instruções Técnicas Virtuais

O sistema de Instruções Técnicas Virtuais, doravante chamado de ITV, é um ambiente de desenvolvimento e visualização de instruções de manutenção e operação de maquinário da Usina Hidrelétrica de Tucuruí, capaz de importar e manipular objetos 3D, gerar animações e definir os métodos de interação do usuário, simulação e execução de uma seqüência de passos em um ambiente 3D interativo.

5.1 Interface GUI e Ambiente 3D

De maneira geral, a interface gráfica do Sistema ITV pode ser dividida em duas principais áreas: o ambiente GUI e o ambiente virtual 3D. A figura 5.1 mostra a interface geral do sistema ITV durante a criação de uma instrução técnica virtual.



Figura 5.1: Interface geral do sistema ITV

O ambiente GUI compreende os componentes que permitem a execução de ações básicas do sistema, como iniciar ou pausar uma animação, inserir novos objetos 3D ou criar mensagens textuais, por exemplo.

O ambiente virtual, que exibe o cenário 3D, possui interatividade própria; ou seja, o usuário pode navegar pelo mundo virtual diretamente, com liberdade total de câmera (movimentos nos três eixos, em quaisquer sentidos) via *mouse* ou teclado, independente dos componentes GUI.

Durante a interação com o ambiente 3D, portanto, o usuário não precisa utilizar os elementos GUI para movimentação de câmera, apesar de isso ser possível. Usando a interface GUI, o usuário pode, entre outras funcionalidades, solicitar que o sistema desloque a câmera automaticamente para uma posição em que um objeto específico seja enquadrado no campo de visão.

5.2 Estrutura Funcional

O funcionamento do sistema ITV utiliza uma metodologia orientada a projetos, que representam as instruções técnicas (doravante chamadas de IT). Um projeto ITV é um artefato criado e executado pelo sistema, que pode ser armazenado em arquivos para posterior utilização. Um projeto ITV contém as informações que constituem uma IT real, considerando os objetos 3D que participam do processo (representando peças, ferramentas, acessórios de segurança, mantenedores, operadores, *etc.*) e as etapas determinísticas que a compõem.

O sistema ITV foi projetado para permitir que diferentes ITs pudessem fazer uso de um mesmo conjunto de objetos 3D, apesar de isso não ser obrigatório. Esse conjunto de objetos funciona como um repositório de *resources* (recursos) compartilhados, e podem ser usados por qualquer IT. Essa característica permite, por exemplo, que uma IT facilmente reutilize um cenário existente em outra IT.

O repositório de recursos (doravante referenciado pela palavra inglesa *media*) é estruturado para permitir que novos objetos sejam incluídos a qualquer momento, tornando-os aptos a participarem de qualquer IT. Dessa forma, o cenário 3D de um projeto ITV é formado por um subconjunto de objetos da *media*, que pode ser expandido a qualquer momento. A figura 5.2 mostra um diagrama que representa as relações entre o aplicativo (*software*), os projetos IT e a *media* (recursos).

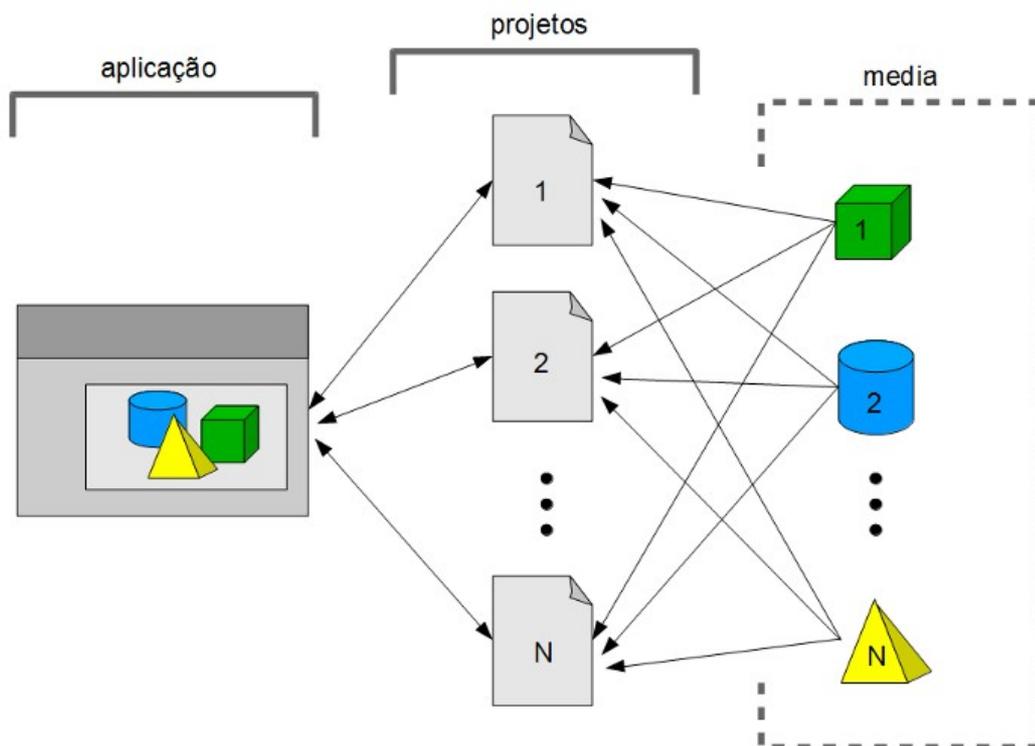


Figura 5.2: Diagrama funcional do sistema ITV

5.3 Arquitetura Computacional

O sistema ITV é composto por um conjunto de tecnologias, que implementam os ambientes virtual e GUI. Entre as principais tecnologias envolvidas, podem ser citadas a linguagem de programação C++, as bibliotecas QT para interfaces GUI, a *engine* gráfica *Ogre3D*, a linguagem XML e os objetos CAD.

5.3.1 Linguagem C++

O sistema ITV foi implementado por meio da linguagem de programação C++, amplamente utilizada em muitas aplicações, com grande índice de popularidade na área de desenvolvimento de softwares. A linguagem C++ caracteriza-se, entre outras peculiaridades, pela metodologia de programação orientada a objetos (*POO*) e pela possibilidade de programação em alto e baixo nível, configurando-se como uma linguagem poderosa e ao mesmo tempo flexível [39]. As bibliotecas QT e *Ogre3D*, usadas na implementação do sistema ITV, são escritas em C++.

5.3.1.1 QT

QT é um pacote de bibliotecas multiplataforma, escritas em C++, voltadas ao desenvolvimento de interfaces GUI. O seu vasto conjunto de componentes GUI possibilita um alto nível de abstração; ou seja, permite que a programação seja focada na implementação das funcionalidades, minimizando o código destinado à interação com o usuário. Estima-se que, em média, 48% do código de uma aplicação se refere apenas à interação com o usuário [40]. Nessa conjuntura, bibliotecas gráficas se mostram muito úteis ao desenvolvimento de aplicações complexas, como o sistema ITV.

5.3.1.2 Ogre3D

Ogre3D é um pacote de bibliotecas, também escrito em C++, destinado ao desenvolvimento de aplicações 3D. Caracteriza-se pela metodologia de orientação a objetos e por atuar como uma camada de *software*, que permite uma abstração de alto-nível, facilitando a utilização das bibliotecas gráficas OpenGL e Direct3D (que exigem uma programação em um mais baixo nível).

O ambiente virtual do sistema ITV é implementado com o Ogre3D, que é, portanto, a camada de *software* que gerencia o cenário 3D; ou seja, manipula os objetos 3D, a câmera, a luz, *etc.*

5.3.2 XML (*eXtensible Markup Language*)

XML é uma metodologia de representação de dados, baseada em texto, cuja sintaxe é padronizada pela W3C (*World Wide Web Consortium*). Um arquivo XML define a estrutura lógica-organizacional dos dados, comumente atribuindo relações de hierarquia entre as entidades. É uma técnica extremamente versátil, usada em diversos tipos de aplicações, tais como páginas de internet, representação de banco de dados, desenhos vetoriais, e muitas outras. Por ser baseada em texto e possuir uma sintaxe padronizada, é muitas vezes usada para intercâmbio de informação entre diferentes aplicações.

De maneira geral, a metodologia XML está presente no sistema ITV em duas situações principais: na representação dos dados de uma IT e na inclusão de objetos 3D no cenário virtual. A estrutura XML que representa uma IT é o próprio artefato que constitui um projeto ITV. Esse arquivo armazena, entre outras informações, os nomes e coordenadas dos objetos 3D (presentes na *media*), tipo de animação (translação, rotação, escala) e seus parâmetros (posição inicial, posição final), posições de câmera, mensagens textuais.

A estrutura XML usada para inclusão de objetos 3D no cenário pode ser considerada como um item auxiliar, uma vez que não é obrigatória. Essa estrutura contém apenas os nomes e coordenadas (de posição e rotação) de uma lista objetos, para que o posicionamento inicial dos mesmos possa ser realizado em lote. Esse arquivo XML, mesmo proporcionando uma confortável usabilidade, pode ser considerado dispensável, já que o usuário pode atribuir uma posição arbitrária a um dado objeto em qualquer momento.

A figura 5.3 mostra um diagrama simbolizando a participação dos dois tipos de estrutura XML utilizadas no sistema ITV: para representar ITs e para auxiliar a importação de novos objetos.

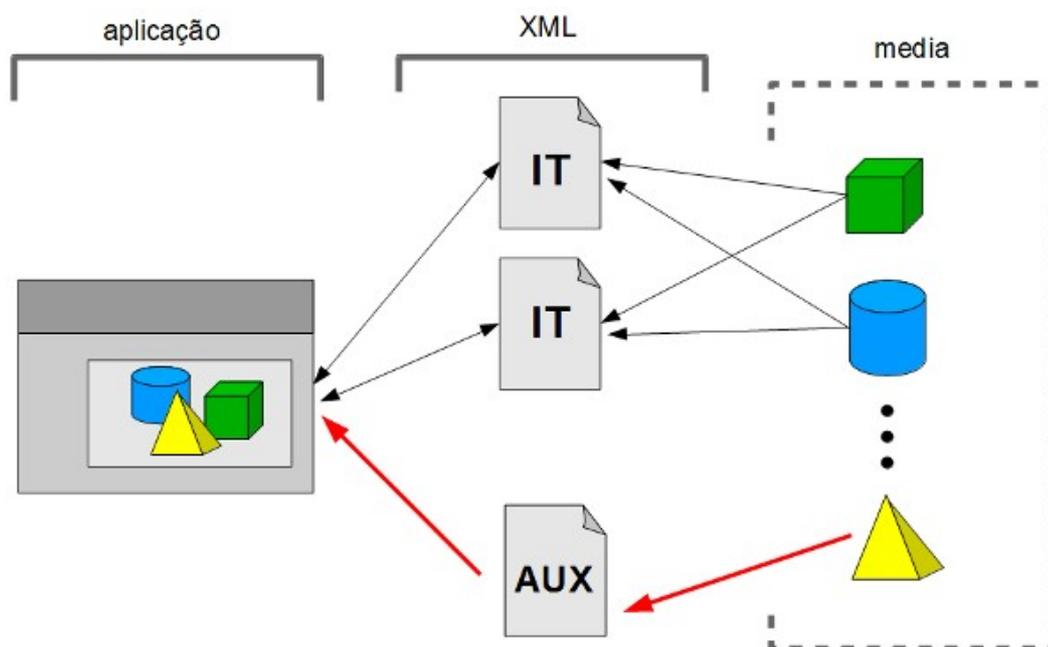


Figura 5.3: Diagrama de utilização das estruturas XML

5.3.3 CAD (*Computer-Aided Design*)

O termo CAD se refere ao uso de tecnologias computacionais como auxílio à criação de projetos gráficos (*design*) 2D ou 3D. É frequentemente aplicado às áreas de arquitetura e engenharia, e por isso está quase sempre associado à construção de desenhos vetoriais.

Desenhos CAD podem ser destinados a muitos tipos de aplicações, e não apenas à ambientes virtuais interativos. Por isso, nesse contexto, o termo *objeto* é geralmente substituído pelo termo *modelo*, uma vez que desenhos CAD são representações (modelos) computacionais de objetos reais. O termo *objeto 3D* é mais utilizado para se referir a um *modelo 3D* inserido no contexto de um cenário tridimensional interativo.

Os objetos 3D utilizados no sistema ITV são, portanto, oriundos de um processo CAD; ou seja, as tecnologias CAD atuam como fornecedoras dos recursos (modelos 3D) que compõem a *media* do sistema. A figura 5.4 mostra um diagrama que apresenta as tecnologias envolvidas e as áreas de influência de cada uma. Como pode ser observado, as tecnologias CAD também são responsáveis por expandir a *media*, criando novos objetos para compor o conjunto de recursos.

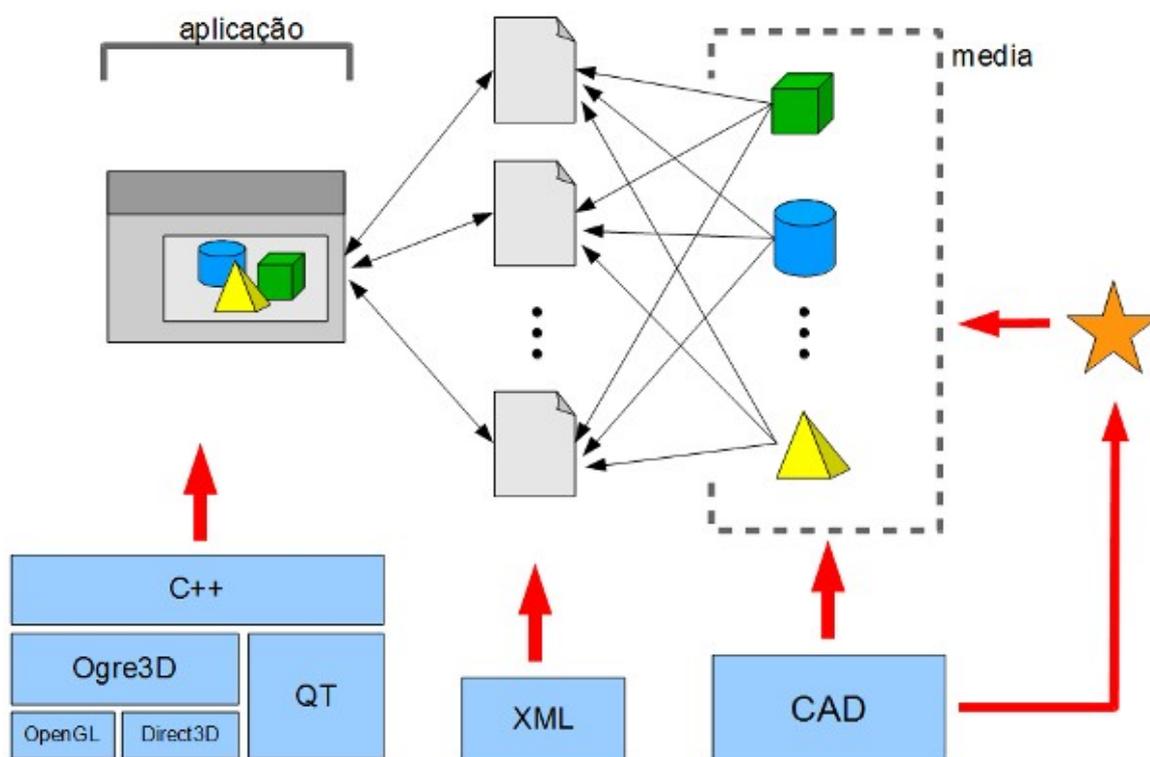


Figura 5.4: Tecnologias envolvidas no sistema ITV e suas áreas de influência

5.4 Organização da *media*

Sendo a *media* um repositório de todos os objetos 3D a serem usados nas ITs, o grande número de elementos que participam desse conjunto e também a possibilidade de expansão exigiram um método de organização cuidadosamente projetado, visando facilitar a inserção de novos elementos, bem como a alteração dos já existentes e compartilhamento desse conjunto pelas diversas ITs.

5.4.1 Arquivos utilizados pelo Ogre3D

Por atuar como a camada de software que gerencia, em alto nível de abstração, o ambiente 3D, o Ogre3D estabelece algumas diretrizes de implementação, principalmente no tocante aos objetos 3D. De forma nativa, o Ogre3D utiliza um padrão específico para representar geometrias, materiais, texturas e os outros elementos do cenário. Os objetos 3D pertencentes à *media* são disponibilizados, portanto, nesse padrão.

5.4.1.1 Geometrias

Nativamente, o Ogre3D utiliza arquivos de extensão *mesh* para representar as geometrias, que nesse caso são sempre baseadas em polígonos. Cada objeto individual possui um arquivo *mesh* que descreve sua geometria. Nesse arquivo também são armazenadas as referências (apenas o nome identificador) aos materiais que esse objeto utiliza. Por ser intrínseco à geometria, o mapeamento (ou seja, as coordenadas de texturas) também é gravado nesse arquivo. Além dessas informações, o arquivo *mesh* armazena ainda o nome da estrutura *skeleton* referente à geometria em questão, se houver.

5.4.1.2 Texturas

Por serem imagens bitmap, as texturas usadas pelo Ogre3D são arquivos em formatos convencionais, tais como PNG, JPG, GIF, TGA, *etc.* O formato PNG, cuja especificação prevê a utilização de um canal *alpha*, é uma opção conveniente para implementar regiões transparentes baseadas em textura.

5.4.1.3 Materiais

Os materiais, que são conjuntos de parâmetros, são gravados como *scripts*; ou seja, são armazenados como um código de instruções textuais, escritas com uma sintaxe específica, a serem interpretados em tempo de execução. Por padrão, esses *scripts* são gravados em arquivos com extensão *material*. Cada material usado num ambiente Ogre3D possui um nome identificador, que o situa, univocamente, entre todos os materiais usados no ambiente. Um arquivo com extensão *material* pode conter a descrição de vários materiais (e não apenas um). Os nomes identificadores independem do nome do arquivo, uma vez que são definidos no *script*. Por esse motivo é que possível armazenar vários materiais em um mesmo arquivo de extensão *material*.

5.4.1.4 Skeletons

As estruturas de esqueleto, que armazenam as animações embutidas desse tipo, são armazenadas em arquivos (de extensão *skeleton*) cujo nome serve como identificador para associar a geometria à estrutura de animação. Ou seja, o arquivo *mesh* armazena o nome do arquivo *skeleton* que descreve suas animações embutidas desse tipo, se existirem. Por relacionar subconjuntos específicos de vértices da geometria diretamente aos ossos (*bones*) do esqueleto, uma estrutura *skeleton* não pode ser compartilhada por qualquer geometria.

5.4.2 Padrão de nomenclatura

Para facilitar o gerenciamento da *media*, foi convencionado um padrão de nomenclatura para seus elementos, incluindo arquivos e outras entidades associadas. De fato, essa nomenclatura não é obrigatória, mas representa uma metodologia de organização que visa uma manipulação eficiente dos recursos.

5.4.2.1 Grupos de objetos

Dado o grande número de objetos, os componentes da *media* foram divididos em grupos lógicos, que correspondem a subdiretórios internos ao diretório geral que representa o repositório.

Por convenção, cada grupo é identificado por uma sigla, de duas a quatro letras, formando um termo mnemônico da descrição do grupo. Ao grupo do *Regulador de Velocidade*, por exemplo, foi atribuída a sigla *RV*. Ao grupo de objetos de concreto, a sigla *CCT*; e assim por diante.

5.4.2.2 Nomes de entidades

Para nomear arquivos, a regra geral adotada foi sempre iniciar o nome partindo da estrutura mais ampla para a mais específica, separando os termos pelo caractere *underline* (em vez de espaço), sempre com letras minúsculas e sem acentuação. Logo, considerando que todos os objetos pertencem a um grupo, o nome do grupo seria sempre o prefixo para o nome do arquivo. Assim, o nome de um parafuso da válvula do regulador de velocidade seria “*rv_valvula_parafuso*”. Essa é a regra geral aplicada aos arquivos *mesh*, às texturas e aos *skeletons*.

No caso dos arquivos de material, dada a possibilidade de utilizar um único arquivo para descrever vários materiais, foi convencionada a utilização de apenas um arquivo de material para toda a *media*. Esse arquivo foi chamado de “*todos.material*”. Uma das vantagens de utilizar um arquivo de material unificado é a obrigação de identificar univocamente os materiais; ou seja, uma vez que não pode haver ambigüidade de nomes de materiais, um arquivo de material que centraliza poderia facilitar a identificação de materiais duplicados.

5.4.3 Estrutura de diretórios

Para cada grupo de objetos existe um respectivo subdiretório da *media*. Todos os arquivos devem estar situados no subdiretório referente ao seu grupo. No caso do arquivo “*todos.material*”, foi criado um subdiretório especial, denominado *comum*, que organiza recursos estruturais e auxiliares do sistema ITV (independentes de IT).

A estrutura de diretórios da *media* não é imutável, e, portanto, novos grupos podem ser criados a qualquer momento conforme a necessidade. De fato, essa organização não é obrigatória; as ITs podem ser executadas independente da estrutura de subdiretórios da *media*.

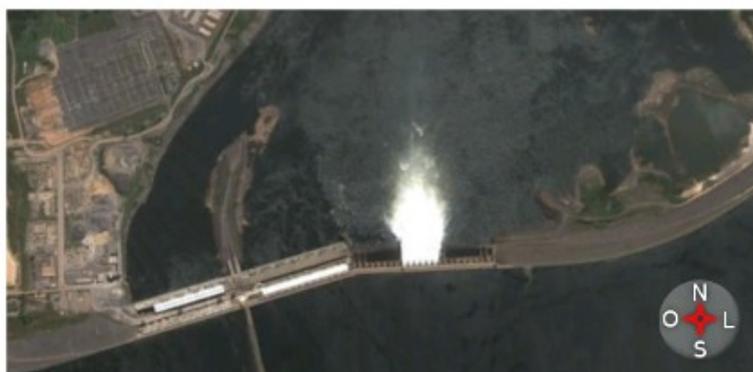
Para um recurso ser utilizado pelo sistema ITV, seja ele uma geometria, textura ou material, é necessário apenas que o mesmo faça parte da *media*, independente de qual subdiretório esteja localizado. É possível, inclusive, que geometria e textura de um mesmo objeto estejam em diferentes subdiretórios. Essa característica funcional permite, além do compartilhamento de objetos por diferentes ITs, também o compartilhamento de texturas por diferentes objetos.

5.5 Contexto da Modelagem 3D

O processo de modelagem 3D para o sistema ITV adotou mecanismos de criação voltados a um gerenciamento eficiente por parte da equipe de modeladores, levando em consideração o cenário alvo da modelagem, as informações disponíveis para tal tarefa e adequada correspondência entre os mundos real e virtual.

5.5.1 Cenário real: instalações da UHE Tucuruí

A usina hidrelétrica (UHE) de Tucuruí possui 23 unidades geradoras hidráulicas (UGH) de grande porte, distribuídas por uma extensão aproximada de 850 metros, e situadas à lateral oeste do vertedouro, que se estende por mais 550 metros, formando a represa no rio Tocantins. O percurso do rio transcorre do lado sul (que representa o lado de *montante*) para o lado norte (lado de *jusante*), estabelecendo uma queda d'água efetiva de aproximadamente 70 metros entre os dois lados. A figura 5.5 mostra uma imagem de satélite com as instalações da usina.



(a) UGHs e Vertedouro



(b) Localização do centro de operação

Figura 5.5: Visão geral da represa da UHE Tucuruí

As UGHs numeradas de 01 a 12 formam o primeiro conjunto de unidades, ao leste do Centro de Operação da Usina (COU). As UGH numeradas de 13 a 23, situadas ao oeste do COU, formam o segundo conjunto, que foi o principal alvo da modelagem 3D. Especificamente, a UGH 13 foi a escolhida para ser detalhadamente representada, sendo o principal cenário das ITs virtuais.

Uma vez que o sistema ITV é projetado para utilizar um conjunto expansível de objetos 3D, não há obrigatoriedade de que apenas a UGH 13 seja representada com modelos virtuais detalhados. De fato, nem mesmo é uma obrigatoriedade que apenas a UHE Tucuruí seja palco das ITs, sendo possível criar ITs de outras usinas, desde que suas peças façam parte da *media* do sistema ITV.

5.5.2 Recursos e Informações Disponíveis

As principais fontes de informação para a modelagem 3D foram os desenhos técnicos 2D (em formato digital), fornecidos pela Eletronorte S/A, bem como centenas de fotografias e vídeos da usina, muitos deles coletados em visitas *in loco* pela equipe de desenvolvimento do sistema ITV.

Os desenhos 2D, disponíveis para a maioria das peças modeladas, possibilitaram a coleta de dados sobre as dimensões de largura, comprimento, espessura, raio, *etc.* Esses desenhos, baseados em vetores, foram usados como principal referência para a criação das geometrias. Por outro lado, dado que esses desenhos representam apenas cortes seccionais dos objetos, as fotografias e vídeos também tiveram importância fundamental para percepção da estrutura tridimensional das peças. As fotografias, em especial, além de contribuir para a definição das geometrias, foram diretamente utilizadas para criação das texturas da grande maioria dos objetos. Os vídeos também contribuíram para determinação das posições dos objetos, proporcionando a percepção espacial do ambiente como um todo, principalmente nos casos em que os desenhos 2D não contemplavam o posicionamento das peças ou equipamentos em questão. Muitos corredores e galerias da usina foram modelados baseados em vídeos, por exemplo.

De forma complementar, muitas informações foram obtidas por meio de diálogos entre a equipe de desenvolvimento do sistema ITV e os funcionários da usina, que forneceram detalhes mais específicos e opinaram sobre o nível de detalhes requerido para cada objeto virtual.

5.5.3 Referências Métricas

Um mundo virtual não possui, efetivamente, uma unidade de comprimento (metro, polegada, etc) para seus eixos. Porém, com o objetivo de tornar mais intuitivo o processo de modelagem e evitar conversões de escala, os valores numéricos do mundo virtual foram associados à unidade *metro* do mundo real.

Alguns dos desenhos 2D de referência representam a planta baixa da usina em uma dada cota (altura) em relação ao nível do mar [41]. Assim, a origem (valor zero) do eixo vertical foi associada ao nível do mar.

A figura 5.6 mostra o desenho 2D que serviu de referência de posicionamento no plano vertical (sentido montante/jusante).

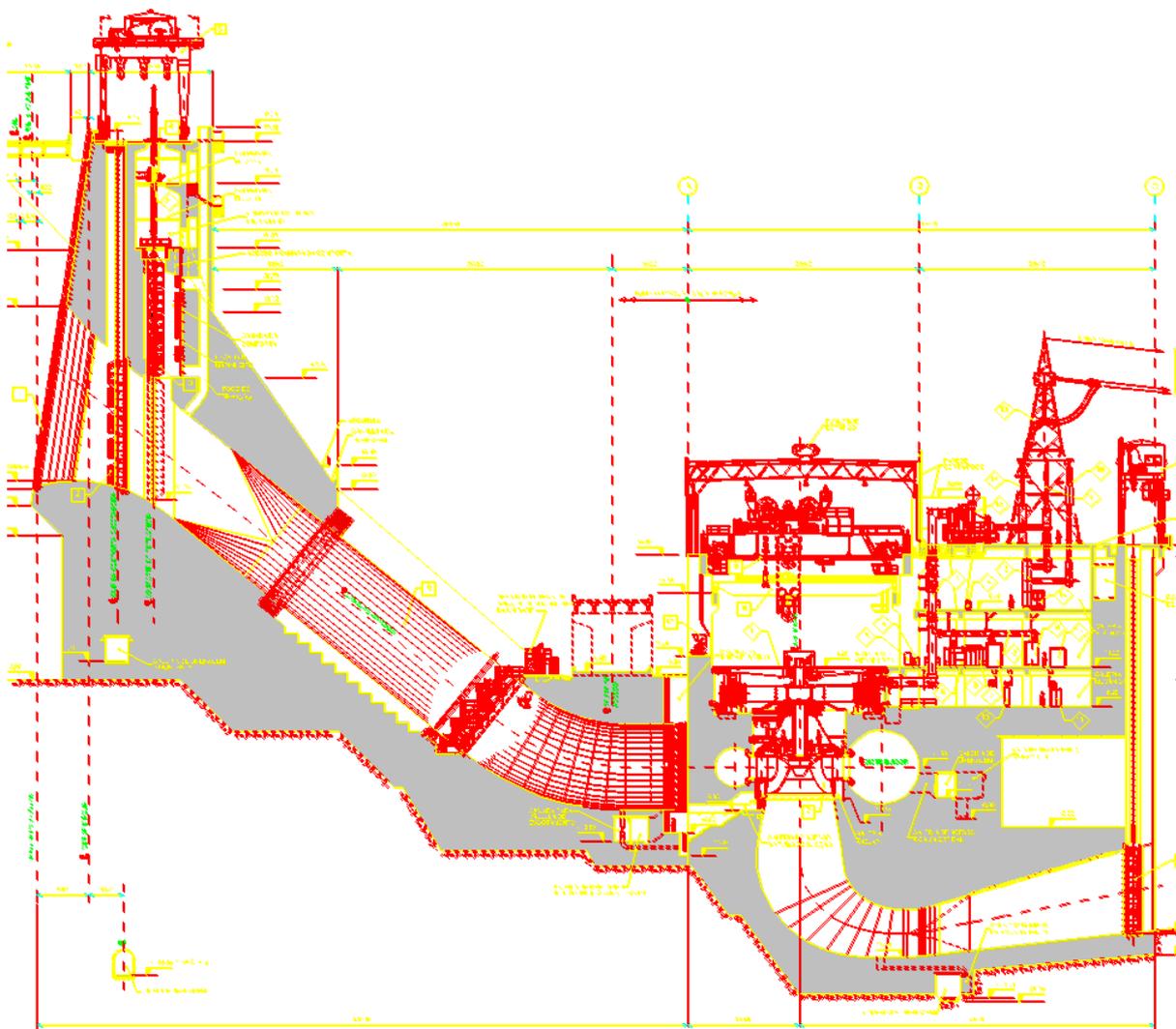


Figura 5.6: Desenho 2D com a visão geral da usina

Considerando que, por convenção, a UGH 13 seria modelada com um nível de detalhes maior que as demais, a origem dos eixos que formam o plano horizontal do mundo virtual foram associados ao centro de rotação do conjunto girante principal (turbina, eixo e rotor do gerador) desta UGH. A figura 5.7 mostra o desenho 2D usado como referência para o plano horizontal.

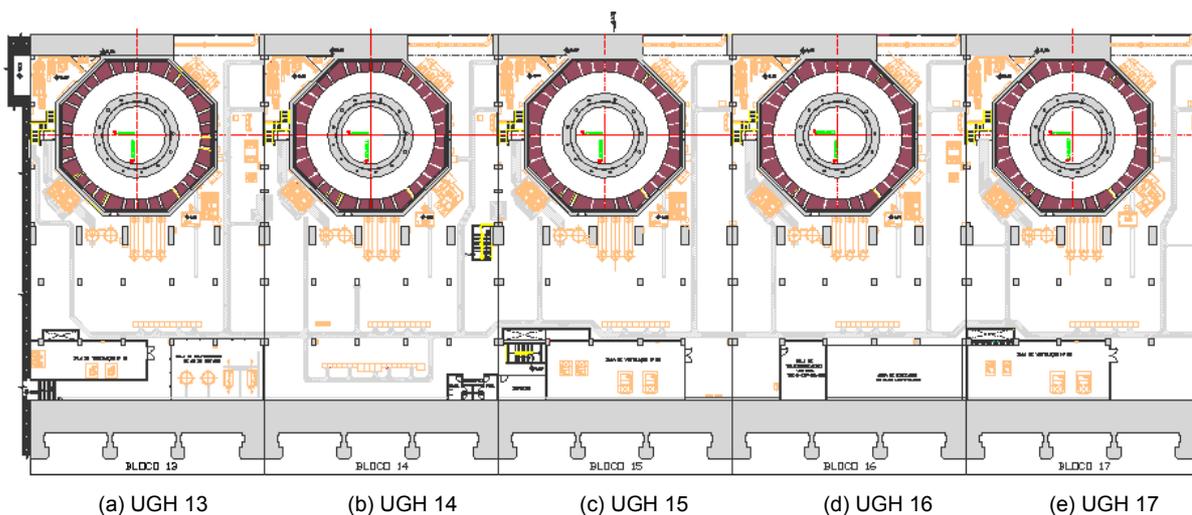


Figura 5.7: Desenho 2D da planta baixa de uma das cotas

A figura 5.7 mostra a relação espacial entre os dois desenhos bidimensionais usados como principal referência para os eixos.

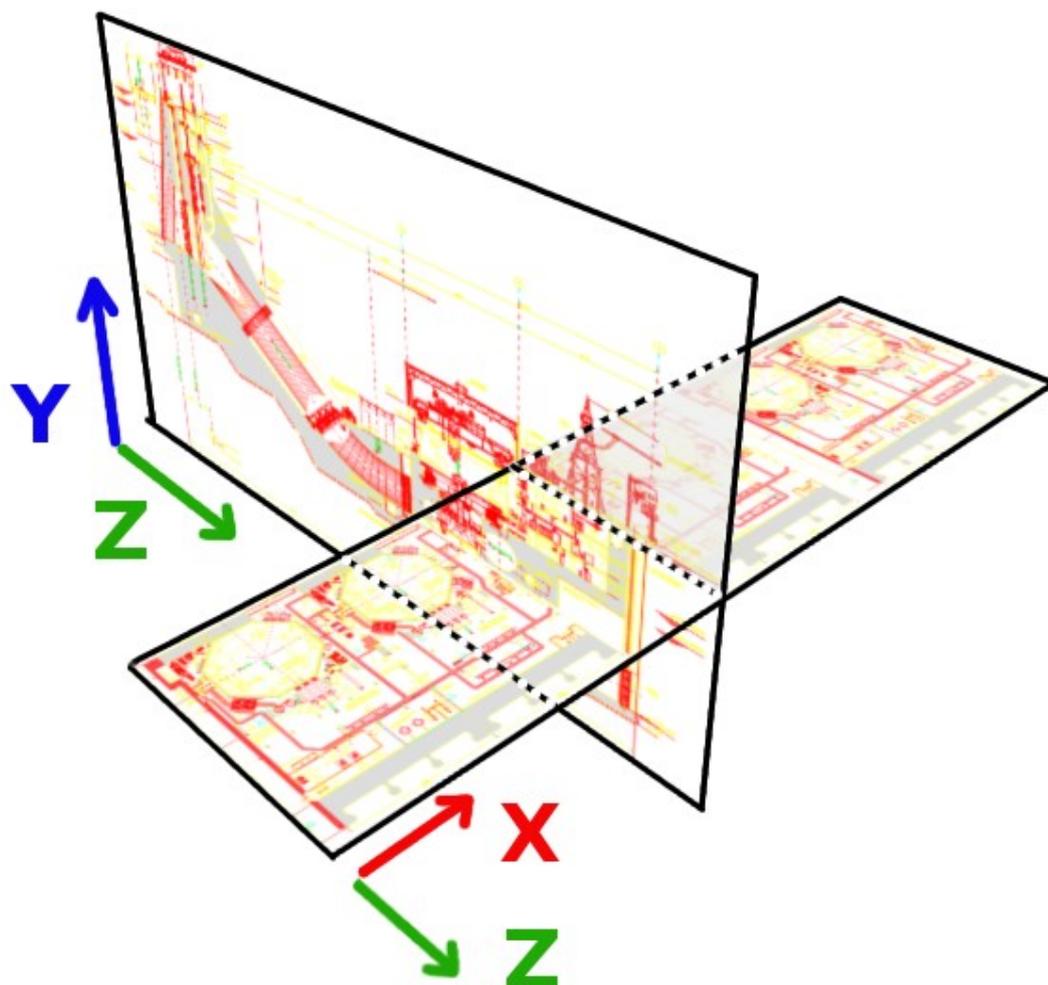


Figura 5.8: Relação entre os desenhos 2D de referência

Essas relações métricas não constituem uma exigência para modelagem 3D, uma vez que os objetos inseridos no mundo virtual podem ter posição e escala alteradas por meio da interface de usuário a qualquer momento. Todavia, essa padronização provê numerosos benefícios ao processo, entre eles a possibilidade de exportar as posições e outras referências métricas diretamente do *software* modelador ao ambiente do sistema ITV, por intermédio do arquivo XML auxiliar. Além disso, essa convenção facilita o compartilhamento de objetos entre diferentes ITs e também auxilia o processo de criação de novos objetos 3D baseados em outros, já existentes ou modelados por outra pessoa.

5.6 Exemplos de objetos 3D criados

Considerando o grande número de objetos que podem participar de uma IT, metodologias foram adotadas para otimizar os objetos e, assim, reduzir o custo de processamento dos cenários 3D, viabilizando interações em tempo-real. O principal procedimento adotado foi construir geometrias com poucos triângulos, sempre que possível. A utilização de texturas foi priorizada, considerando o menor custo computacional que elas podem exigir se comparadas às geometrias com muitos polígonos.

Para cada caso, foi estabelecido um *trade-off*; ou seja, uma ponderação entre nível de detalhes requerido, aspecto visual realístico, relevância do objeto e necessidade de visualização em vários ângulos de câmera. Para alguns objetos, foi criada mais de uma versão, com diferentes quantidades de polígonos, adequadas a cada IT.

Objetos projetados para serem fixos (sem necessidade de animação, com ângulo de visualização limitado ou com pouca percepção de perspectiva) foram criados com uma geometria detalhada apenas nas regiões visíveis, e simplificada nas regiões ocultas. Em alguns casos, a geometria desses objetos era composta por apenas dois triângulos, formando um quadrilátero texturizado.

Sempre que viável à organização da *media*, os objetos eram unificados por questões de desempenho. Dados dois cenários com as mesmas quantidades de triângulos, o cenário com menor número de objetos tende a obter um melhor desempenho. Assim, em situações convenientes, as geometrias de objetos fixos eram unificadas em uma única geometria.

5.6.1 Terreno

Uma técnica muito utilizada para representar terrenos, conhecida como *heightmap*, consiste determinar a geometria poligonal do terreno com auxílio de uma imagem bitmap, que representa um *mapa de altura*. A altura do terreno em cada região é determinada de acordo com as tonalidades da imagem na área correspondente, de forma similar aos mapas hipsométricos mostrados na figura 3.7.

A implementação da técnica *heightmap* disponibilizada de forma nativa pelo Ogre3D possui um algoritmo de otimização que divide a geometria em partes, para que a geometria seja processada de acordo a campo de visão da câmera. Essa técnica de otimização oferece ganhos significativos de desempenho para terrenos muito irregulares.

Como as ITs ocorrem dentro do ambiente virtual da usina, o objetivo do terreno externo era apenas propiciar uma visão panorâmica da região (sem detalhes relevantes), a geometria do terreno demandaria poucos polígonos. Por esse motivo, a geometria do terreno foi construída da forma convencional; ou seja, manualmente desenhada no software modelagem 3D, sendo possível, assim, escolher quais áreas da geometria precisariam ter uma maior densidade de polígonos.

Como auxílio à construção da geometria, foi adotada a técnica de contorno da referência visual, na qual uma imagem bitmap (que representa o objeto real, geralmente uma fotografia) é exibida como *background* na área de desenho, para que seja contornada. A linha de contorno é então convertida para uma superfície 3D.

No caso do terreno, a referência foi uma imagem de satélite obtida pela aplicação *web Landsat* [42], contemplando uma área aproximada de 64 Km² ao redor da usina. Apesar de ser o maior objeto (em extensão) do cenário virtual, a geometria do terreno foi construída com apenas 427 triângulos, menos que o número utilizado na geometria da chave inglesa mostrada na figura 5.7 (com 464 triângulos). A textura aplicada à geometria foi a própria imagem de referência (devida editada), configurando um processo de criação de geometria baseada em textura. A figura 5.9 a imagem usada com referência (a) e as geometrias.

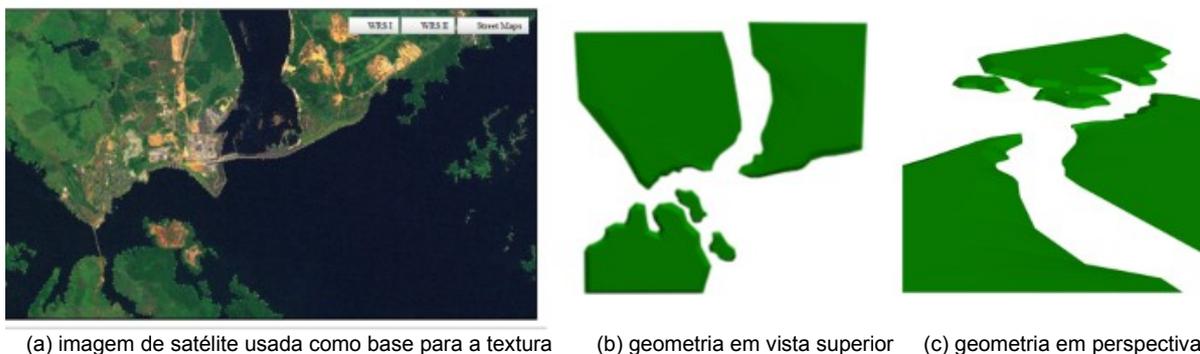


Figura 5.9: Foto de referência e geometrias do terreno

A figura 5.10 mostra dois cenários com o terreno no ambiente virtual do sistema ITV.

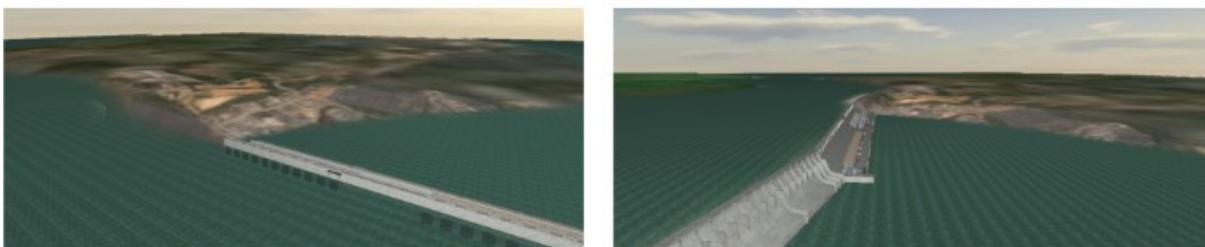


Figura 5.10: Imagens do terreno no ambiente virtual do sistema ITV

5.6.2 Água

Animações virtuais de água podem ser implementadas por sistemas de partículas, que são indicadas para situações em que sejam necessários cálculos de colisão, como preenchimento volumétrico de um determinado recipiente.

No caso da água do rio Tocantins, como o objetivo principal era visualizar a diferença de nível entre montante e jusante, não foram usadas partículas. Além disso, considerando a grande área que deveria ser coberta por água, o uso de partículas seria inviável, dado o grande custo computacional que um sistema com muitas partículas pode demandar.

Para simular a dinâmica da água, foi usado um material configurado para movimentar (deslocar continuamente) as texturas. Para aumentar o realismo do movimento, utilizou-se uma combinação de texturas sobrepostas, que, de fato, são a mesma textura com movimentos em diferentes velocidades. A geometria da água foi implementada com poucos polígonos, uma vez que o contorno do rio foi determinado por meio de uma máscara. Para representar a diferença de nível, os polígonos foram divididos em duas principais regiões: antes da represa (montante) e depois da represa (jusante).

A figura 5.11 as texturas e máscaras utilizadas e o resultado visual obtido no ambiente virtual.

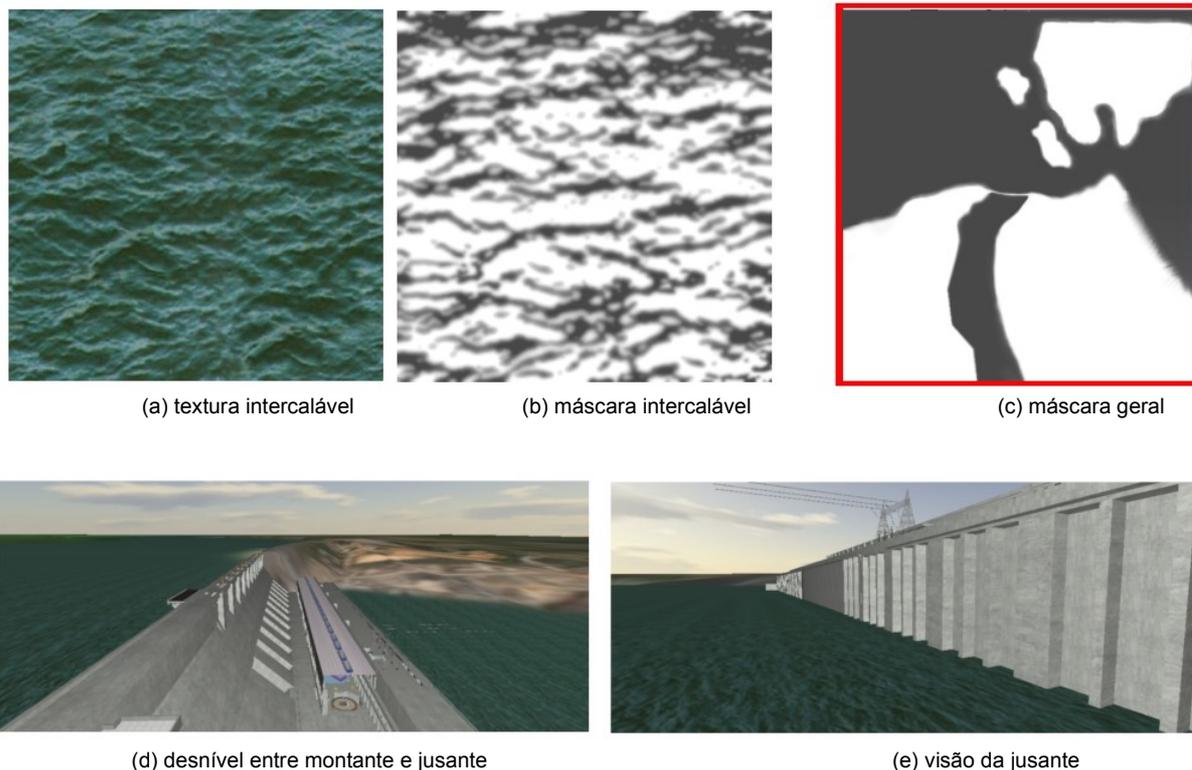


Figura 5.11: Imagens da água no ambiente virtual do sistema ITV

5.6.3 Concretos

A maioria dos elementos de que representam a construção civil foram construídos com base nos desenhos 2D (plantas). Dada a grande quantidade de escadas, paredes, pisos e colunas de concreto, a construção civil foi dividida em objetos 3D separados, para proporcionar uma modularidade e facilitar possíveis alterações nesses objetos; ou seja, essas geometrias não foram unificadas. Por outro lado, mesmo com geometrias independentes, esses objetos compartilham um mesmo conjunto de material e texturas.

Considerando que a construção civil é de concreto, todos esses objetos utilizaram a mesma textura. Porém, alguns elementos de concreto precisavam obter uma aparência mais escura do que os outros, dependendo do local da usina. Foram criados então, dois materiais com diferentes intensidades de brilho (mas utilizando a mesma textura).

No caso das 10 últimas UGHs (numeradas de 14 a 23) da usina, os objetos de concreto foram construídos de modo particular. Essas UGHs foram modeladas com geometrias simplificadas, já que as instruções técnicas seriam executadas sempre (por mera convenção) na UGH 13. Pelo mesmo motivo, esses objetos de concreto foram unificados (formando uma mesma geometria) a outros elementos (piso da plataforma superior, por exemplo) e construídos com um recurso adicional nas texturas. Para simplificar a geometria das 10 últimas UGHs, utilizou-se máscaras para fazer a combinação de texturas e para representar cavidades no concreto. A figura 5.12 mostra a textura utilizada para os objetos de concreto (a) e a simplificação realizada nas UGH de 14 a 23 (b) com relação à UGH 13 (c).

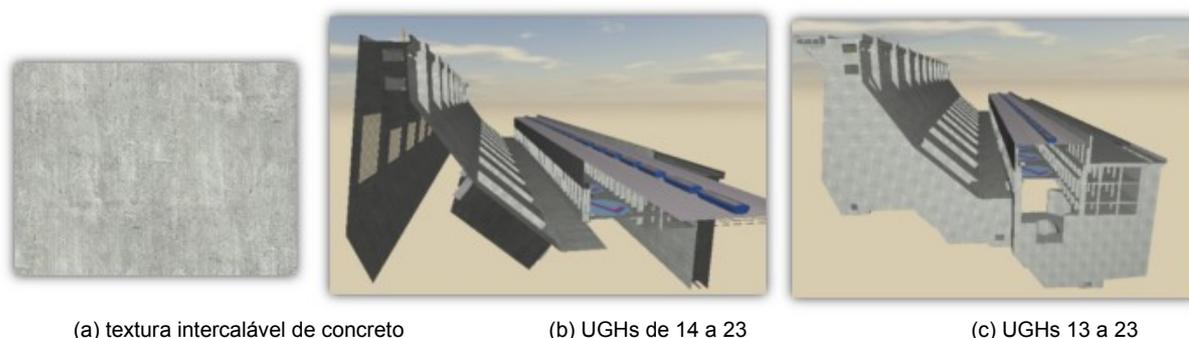


Figura 5.12: Objetos de concreto do sistema ITV

5.6.4 Rotor do Gerador (versão simplificada)

Por questões de desempenho, alguns objetos tinham duas versões: uma detalhada e uma simplificada. A versão detalhada, construída com mais polígonos, é destinada às ITs em que o objeto participa da montagem ou desmontagem. A versão simplificada é usada em ITs cujos cenários exibem o objeto sem efetiva participação nos processos de montagem.

Na IT de manutenção do gerador, o rotor foi modelado com o nível de detalhes adequados aos requisitos visuais dessa instrução, já que o rotor era uma das principais peças envolvidas. Para as outras ITs foi criada uma versão otimizada do rotor, de modo a proporcionar um aspecto realístico, mas sem a possibilidade de ser desmontado. A principal otimização do rotor ocorreu na parte central, conhecida como *aranha* do rotor.

As aberturas que formam uma região circular foram implementadas por máscara. Devido a grande dimensão do rotor (em relação a outros objetos), a máscara utilizada representava um semi-círculo, que ao ser “espelhado” (replicado de forma simétrica) forma o círculo completo. O recurso de espelhamento permite utilizar uma imagem (máscara ou textura) com metade da resolução, já que a outra metade é simétrica. A figura 5.13 mostra a geometria (a), uma das máscaras utilizadas (c) e o resultado visual no ambiente do sistema ITV (d).

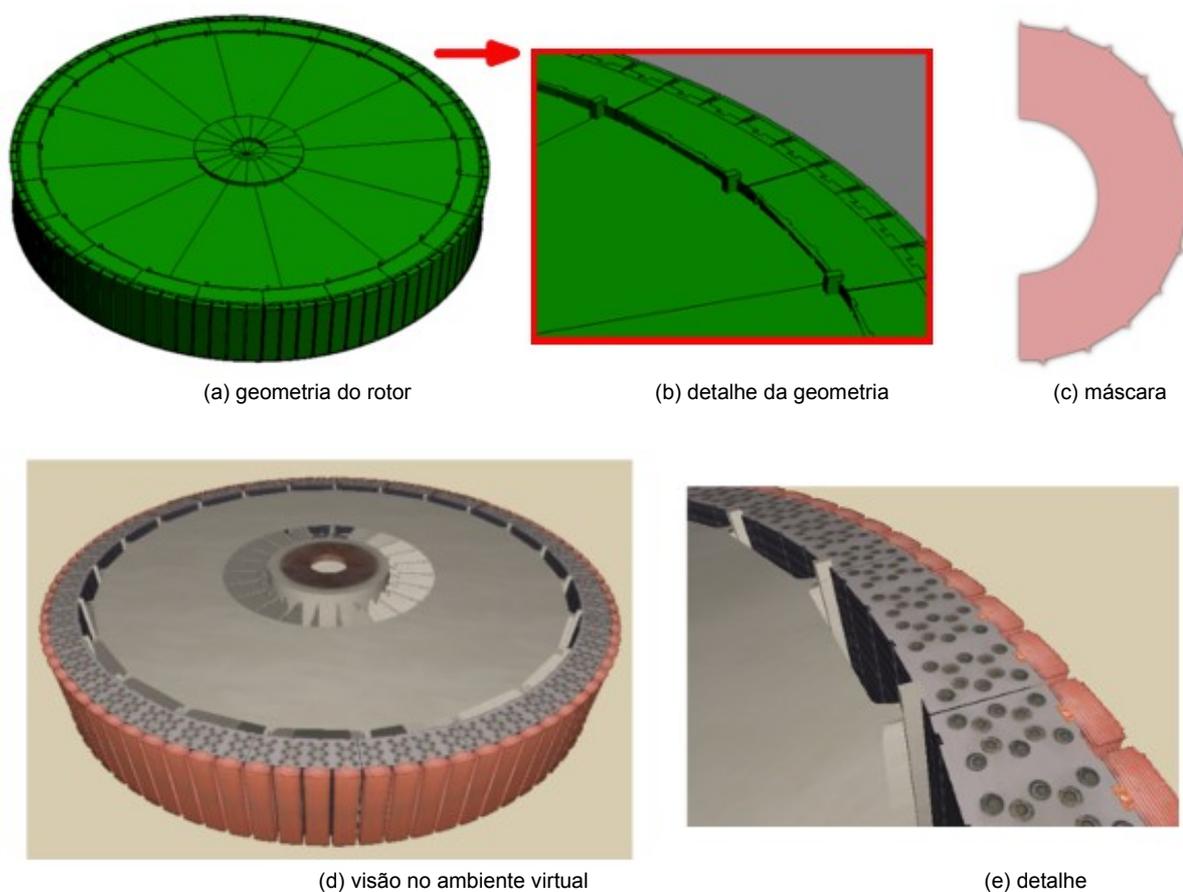


Figura 5.13: Rotor do gerador no ambiente virtual

5.6.5 Tubulações com setas animadas

Durante a execução das ITs, para simbolizar fluxos de água (ou óleo) dentro de tubulações, são mostradas setas animadas. Esse tipo de animação pode ser criada pelo material do objeto; ou seja, configurando, no material, o deslocamento contínuo da textura (como realizado com a animação da água).

Uma característica particular das setas animadas é a geometria simplificada, que consiste em seções retangulares (ao redor da tubulação efetiva) e texturas (representando as setas) com transparência. O sentido (orientação) das setas era determinado pelo mapeamento (o material determinava apenas a animação).

As geometrias das setas precisavam ser diferentes para cada tubulação, já que as tubulações também eram diferentes. Os materiais e texturas, porém, puderam ser compartilhados de acordo com as possibilidades de cada caso. Como foi convenicionado que tubulações de óleo teriam setas amarelas, por exemplo, todos esses objetos poderiam utilizar a mesma textura de seta. Como o material das setas utiliza animações de textura, somente seria preciso utilizar materiais diferentes nos casos de vazão (velocidade de fluxo) diferente. A figura 5.14 mostra a textura (com transparência) que representa a seta e o efeito visual obtido (b).

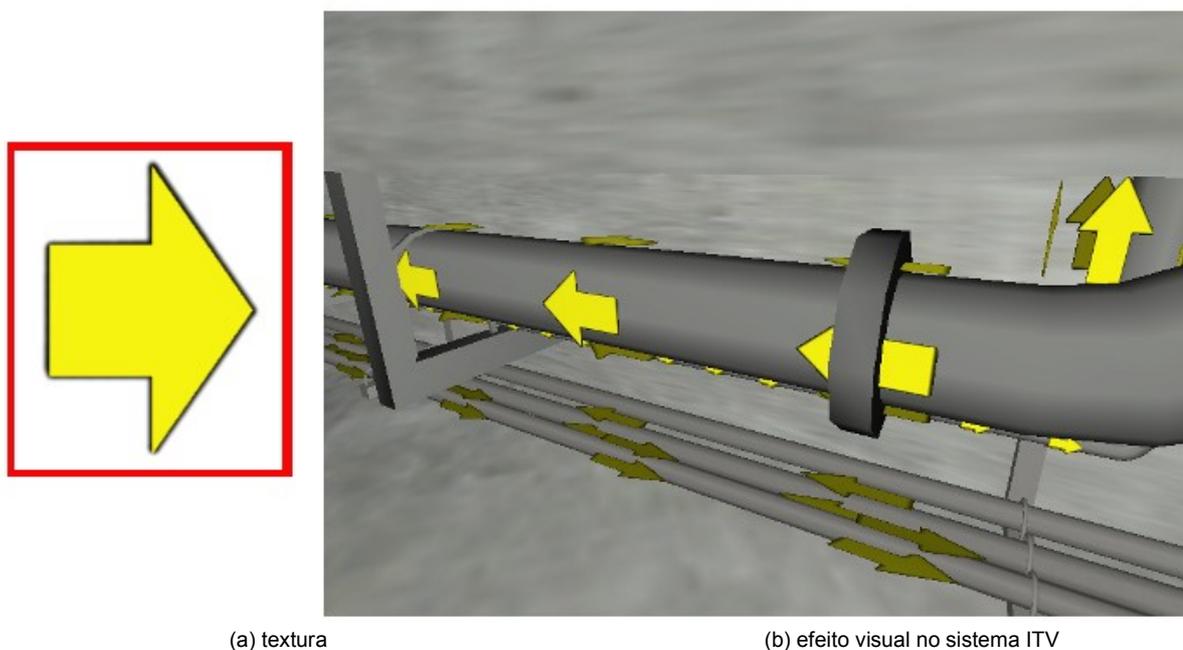


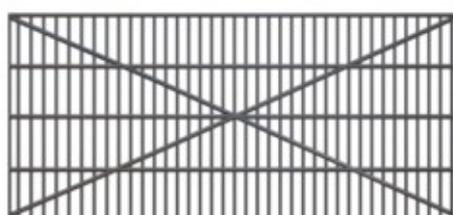
Figura 5.14: Tubulações com setas indicativas de fluxo

5.6.6 Grades

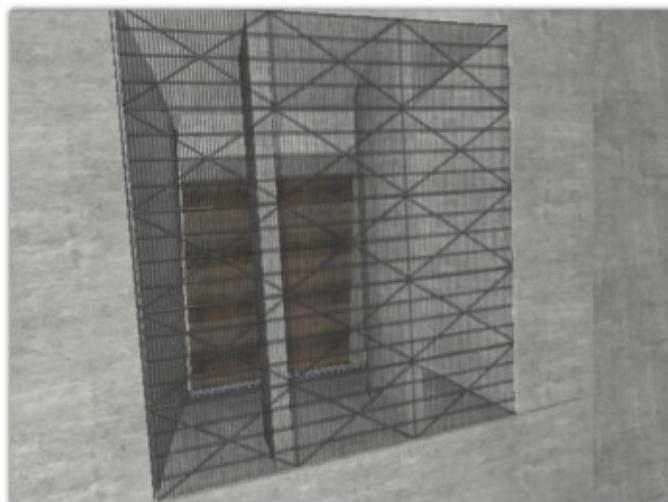
Grande parte das grades metálicas do ambiente virtual foram criadas com geometrias simplificadas, principalmente com relação à espessura. Nesses casos, as barras de metal que compõem as grades foram implementadas por texturas transparentes.

5.6.6.1 Grade da tomada d'água

A geometria da grade da tomada d'água foi modelado com apenas 2 triângulos. Devido a grande dimensão da grade e significativa densidade de barras, foi utilizada uma textura intercalável para representar as barras. A figura 5.15 mostra a textura (a) utilizada e o resultado visual obtido (b).



(a) textura com canal *alpha*

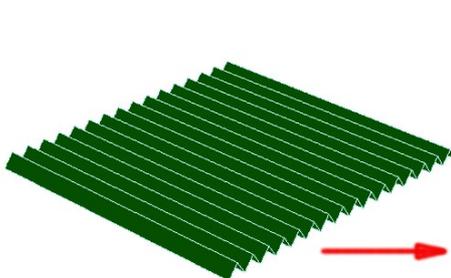


(b) cenário no sistema ITV

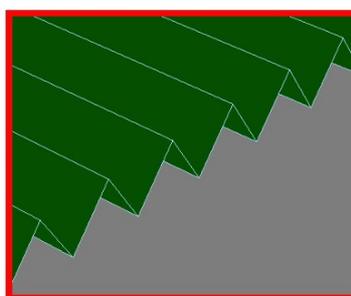
Figura 5.15: Grades da tomada d'água (montante)

5.6.6.2 Grades do teto

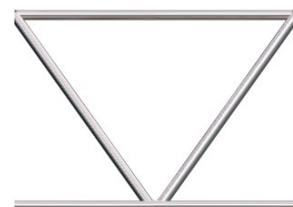
A geometria das grades do teto foram compostas por várias fileiras (com 2 triângulos cada) com inclinações simétricas alternadamente. Dada a grande densidade de barras, também foi necessária uma textura intercalável. A figura 5.16 mostra a geometria (a) e a textura intercalável com canal *alpha* (c).



(a) geometria



(b) detalhe da geometria



(c) textura intercalável

Figura 5.16: Geometria e texturas da grade do teto

A figura 5.17 mostra uma fotografia do salão de máquinas durante a descida de uma das turbinas (a) e uma ampliação mostrando o formato das grades do teto. Em (b) é mostrado o resultado obtido no ambiente virtual.

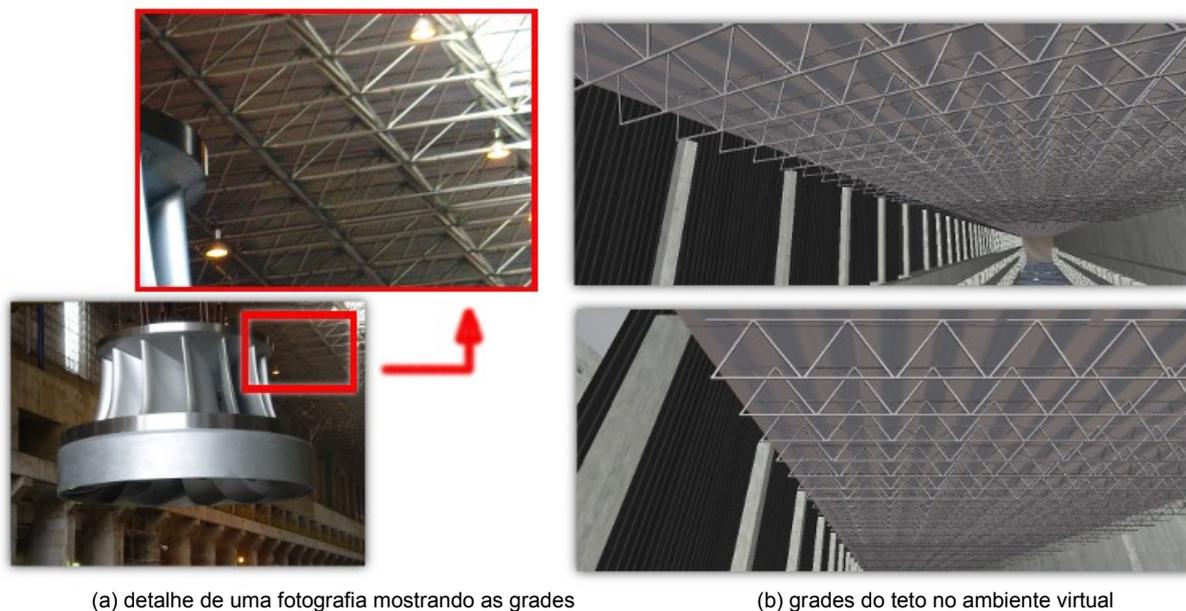


Figura 5.17: Fotografia do salão de máquinas e sua representação virtual

5.6.6.3 Grades de proteção

Algumas grades, com dimensões menores e pequena densidade de barras, utilizaram texturas não intercaláveis. Nesses casos, a vantagem de usar texturas não-intercaláveis é a possibilidade de representar barras de diferentes espessuras com uma única textura. A figura 5.18 mostra a textura de uma grade (a) e o resultado visual obtido no ambiente virtual.

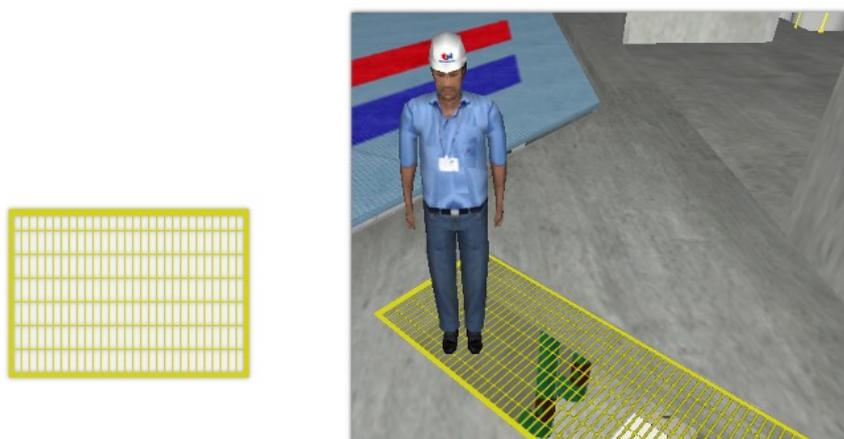


Figura 5.18: Grade no piso do salão de máquinas

5.6.7 Manômetros e ponteiros

As geometrias dos manômetros (e outros dispositivos de medição similares) foram construídas sem ponteiros, para que todas as animações envolvendo esses objetos pudessem utilizar um ponteiro compartilhado.

Os ponteiros, por possuírem uma espessura irrelevante e sempre estarem posicionados dentro dos manômetros, puderam ser construídos com um geometria simplificada (com apenas 2 triângulos) e textura com transparência. Uma vez que o geometria poderia ser a mesma para todos os ponteiros, foi utilizado apenas um material e, conseqüentemente, a mesma textura para todos os ponteiros. A figura 5.19 mostra a textura dos manômetros e ponteiros (a) e o resultado visual (b).



(a) textura com canal *alpha*

(b) resultado visual no ambiente virtual

Figura 5.19: Textura dos manômetros

5.6.8 Centro de Operação da Usina

Algumas etapas das ITs, principalmente as instruções de operação, mostram a sala do Centro de Operação da Usina (COU), com ênfase às telas exibidas nos monitores de vídeo virtuais. Para dar mais realismo ao ambiente, outros elementos também foram modelados, de forma simplificada.

5.6.8.1 Telas dos monitores

Alguns dos monitores virtuais mostram telas que possuem interação com o usuário (indicando as etapas reais), e outros mostram telas alternadas apenas para simbolizar a utilização. As telas exibidas nesses monitores são imagens (texturas) obtidas do próprio sistema de controle da usina, devidamente editadas em alguns casos.

Nos monitores virtuais que possuem interação, o usuário pode utilizar o *mouse* para clicar nas telas e ver os efeitos de sua ação, como se estivesse utilizando o computador virtual e, conseqüentemente, realizando os procedimentos de controle. Cada tela virtual envolvida numa interação é um objeto diferente, com geometrias independentes e texturas diferentes.

Os monitores que mostram telas alternadamente, sem interação, são formados por uma única geometria e único material. Cada tela exibida nesses monitores também constitui uma textura diferente, mas, nesses casos, as transições de tela são realizadas pelo material, semelhante às animações das setas (mas sem deslocamento, apenas transição). A figura 5.20 mostra a sala de comando exibida no ambiente virtual.



Figura 5.20: Monitores de vídeo da sala de comando

5.6.8.2 Corredor

O corredor de acesso a alguns painéis de controle possui uma iluminação característica, que poderia ser emulada por luzes dinâmicas no ambiente virtual. A utilização de luzes dinâmicas é indicada implementar sombras de acordo com os movimentos dos objetos. As sombras do corredor não foram criadas por luzes dinâmicas, e sim por duas texturas, combinadas por máscaras. A figura 5.21 mostra o corredor no ambiente virtual.



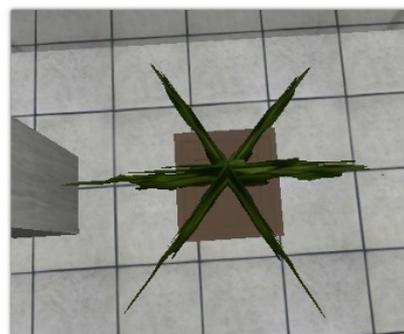
Figura 5.21: Iluminação do corredor por meio de texturas

5.6.8.3 Vasos de plantas

As plantas que ornamentam o COU e a sala de espera foram criadas com texturas transparentes aplicadas a quadriláteros, contabilizando 12 triângulos para o vaso da sala de espera e 16 para o vaso da sala de comando. A figura 5.22 mostra que essa técnica apresenta um resultado satisfatório para ângulos de visão lateral.



(a) ângulo de visão convencional



(b) visão superior

Figura 5.22: Vaso de planta no sala de comando

5.6.9 Parafusos

Parafusos estão presentes em grande parte dos cenários de montagem e desmontagem. Considerando as diferentes espessuras e tamanhos de parafusos para cada situação, não seria possível usar o mesmo objeto (como ocorreu com o ponteiro dos manômetros) para todos os casos. Por outro lado, a textura usada para representar as roscas foi compartilhada por vários parafusos. Com relação aos materiais, não foi utilizado um único material compartilhado; e sim um material por grupo, por motivos de organização (diferentes dos concretos, os parafusos poderiam pertencer a diversos grupos). A figura 5.23 mostra a textura de rosca (a) e o encaixe perfeito quando replicada.



Figura 5.23: Textura de rosca de parafusos

5.6.10 Tampa Superior

A tampa superior do sistema distribuidor de uma UGH é composta por 24 furos circulares, destinados ao encaixe das palhetas diretrizes. Geometrias com curvas sempre exigem um número significativo de triângulos e, por isso, a principal otimização da tampa superior ocorreu nesses 24 furos. Os furos circulares superiores foram modelados com a quantidade de polígonos necessária para representar as curvas com a fidelidade requerida. Os furos inferiores foram representados por texturas com transparência. A figura 5.24 mostra a parte inferior do objeto.

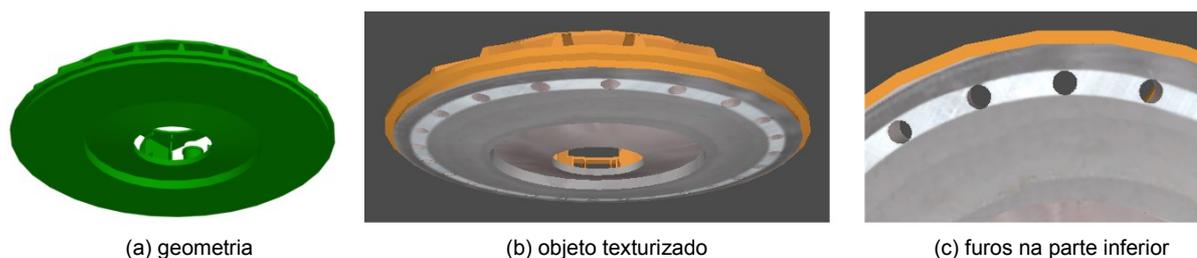


Figura 5.24: Tampa superior do sistema distribuidor (furos inferiores)

A figura 5.25 mostra que a base da geometria não possui furos, mas, quando texturizada, as regiões transparentes simulam os furos na parte inferior.



(a) geometria

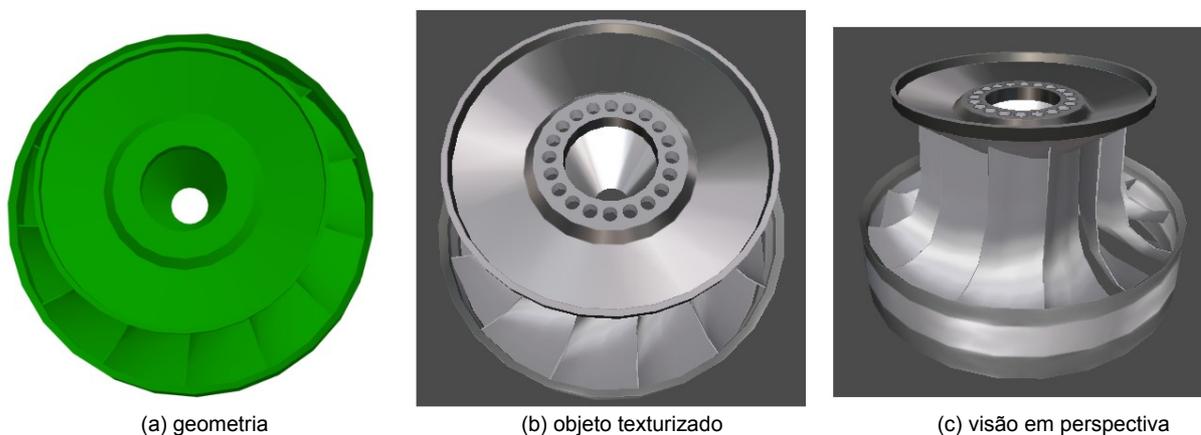
(b) objeto texturizado

(c) visão em perspectiva

Figura 5.25: Tampa superior do sistema distribuidor (furos superiores)

5.6.11 Turbina Francis

A técnica de otimização aplicada à roda da turbina (*francis*) foi semelhante à usada na tampa do sistema distribuidor. Os furos da superfície superior da turbina foram representados por texturas com transparência. Para evitar que fosse percebida a natureza *boundary* da geometria, foram usados polígonos auxiliares internos à geometria, evitando que mais triângulos fossem necessários para a superfície superior. A figura 5.26 mostra a geometria e o resultado obtido.



(a) geometria

(b) objeto texturizado

(c) visão em perspectiva

Figura 5.26: Rotor da turbina

5.6.12 Escotilha da caixa espiral

A escotilha da caixa espiral, que permite acesso à sua galeria, possui um sistema de abertura articulado, cuja animação não seria de fácil implementação pelo módulo de autoria do sistema ITV. Por isso, foi utilizada uma animação *skeleton* com *bones* articulados para abertura e fechamento da escotilha. Os furos destinados aos parafusos foram representados por texturas. A figura 5.27 mostra a escotilha no ambiente virtual.

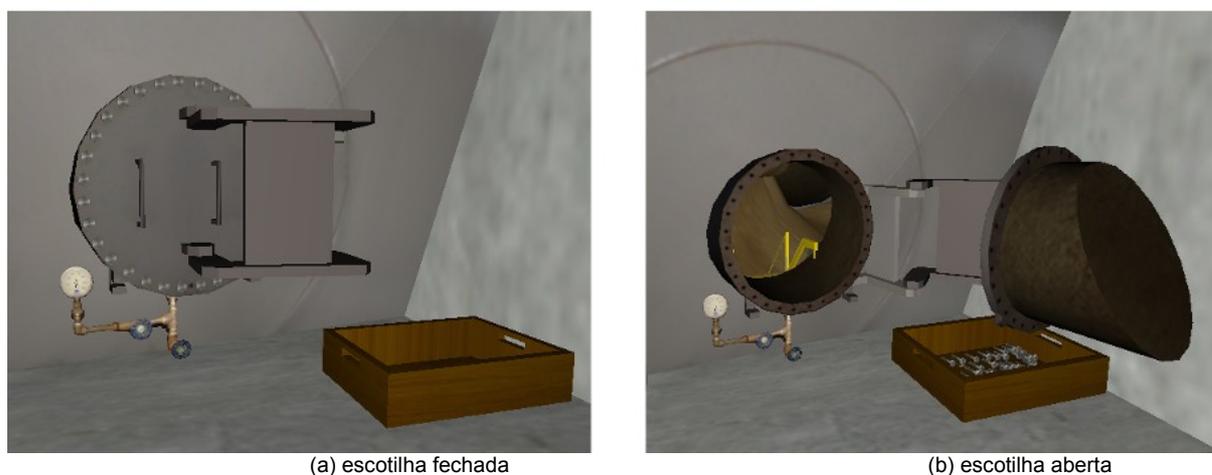


Figura 5.27: Escotilha de acesso à caixa espiral

5.6.13 Molas

Movimentos de extensão e encolhimento de molas podem ser implementados por animações de escala no eixo desejado. Esse tipo de animação de mola apresenta efeitos visuais satisfatórios para pequenos movimentos de extensão ou encolhimento.

5.6.13.1 Válvula detectora de pressão

A mola da válvula detectora pode ser estendida por mais que o dobro do seu comprimento original (não-tensionada). Esse comportamento exige um método de animação mais elaborado, e por isso essa mola foi criada com uma animação *skeleton*. Diferente da escotilha, os *bones* usados na mola não possuem articulação e são distribuídos alinhados à espiral, cujas espiras foram, de fato, representadas por polígonos. A figura 5.28 mostra a mola totalmente encolhida (a) e parcialmente estendida (b), mostrando a geometria em *wireframe* (imagens inferiores).

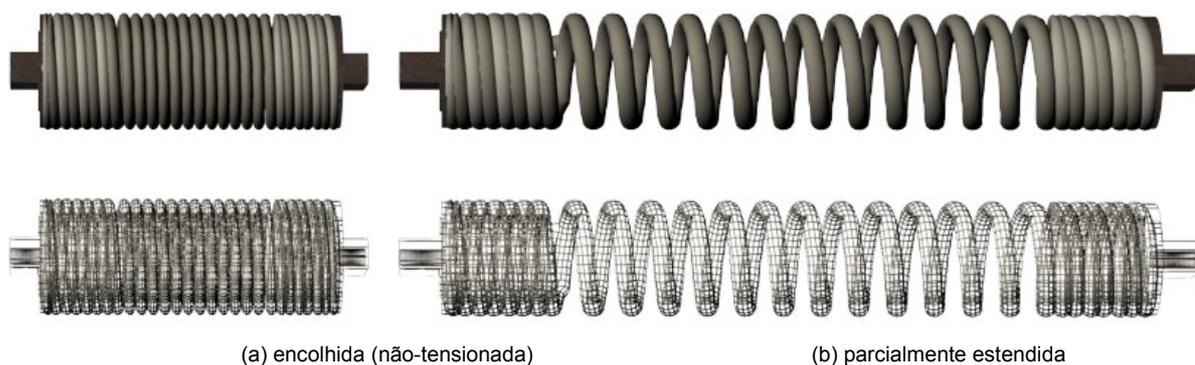
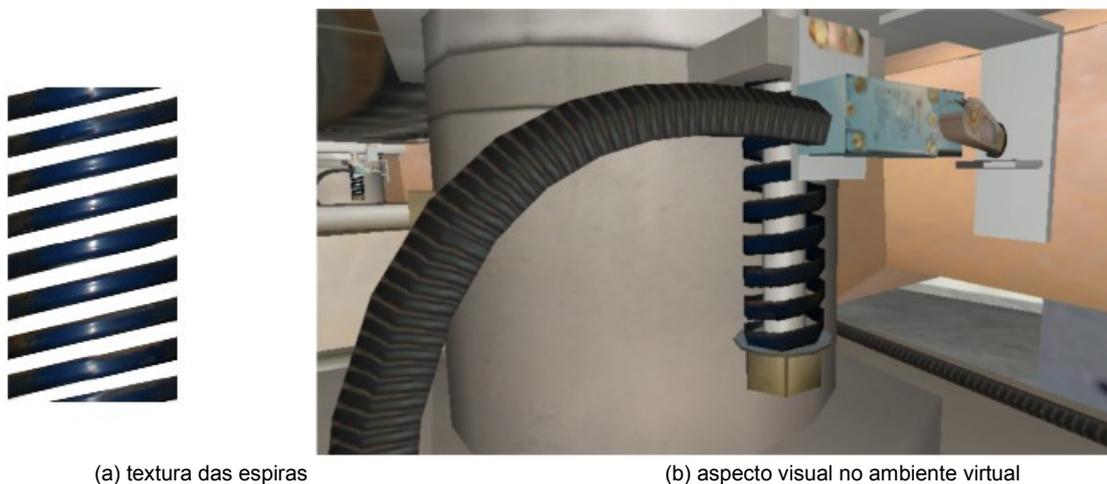


Figura 5.28: Mola da válvula detectora de pressão (objeto texturizado e visão em *wireframe*)

5.6.13.2 Sistema de frenagem

A mola do sistema de frenagem possui um movimento muito sutil, e por isso sua animação foi feita apenas alterando sua escala no eixo vertical. Assim, foi possível simplificar a geometria da espiral (construída como se fosse um cilindro), cujas espiras foram representas por texturas intercaláveis com transparência (com mesmo estilo de encaixe da rosca dos parafusos). A figura 5.29 mostra a textura intercalável com canal *alpha* (a) usada para as espiras da mola e o resultado visual no ambiente virtual (b).



(a) textura das espiras

(b) aspecto visual no ambiente virtual

Figura 5.29: Mola do sistema de frenagem

Capítulo 6 – Conclusão

Este trabalho abordou conceitos e técnicas de representação computacional que servem de alicerce para a implementação de ambientes virtuais e informações visuais de um modo geral, com ênfase à criação e manipulação dos objetos tridimensionais e as tecnologias que os compõem, comentando as formas de representação comumente utilizadas e, em particular, com relação aos objetos destinados a ambientes interativos, que exigem métodos mais elaborados para representar o mundo real, ponderando as limitações de processamento e aspecto visual realístico.

As características de ambientes tridimensionais típicos foram comentadas, contextualizando seus elementos e a participação dos mesmos no processo de interação com o usuário. Entre esses elementos, os objetos 3D foram abordados mais profundamente, citando algumas técnicas de representação geométrica e os princípios computacionais acerca deles.

A técnica de representação de objetos 3D realísticos baseados em polígonos texturizados constituiu um dos principais focos deste trabalho, uma vez que esses objetos participam diretamente do processo de construção de ambientes tridimensionais interativos, tais como jogos eletrônicos e outras aplicações que exigem interação e respostas em tempo-real. Foi dado destaque aos principais elementos que compõem esse tipo de objeto e como eles se relacionam entre si, considerando a possibilidade de compartilhamento desses recursos entre diferentes objetos inseridos no contexto de um ambiente virtual genérico.

O sistema de Instruções Técnicas Virtuais (ITV), destinado a treinamentos e simulações visuais da Usina Hidrelétrica de Tucuruí, foi utilizado como estudo de caso, provendo exemplos não apenas sobre a criação de alguns dos objetos 3D, como também da estrutura organizacional desses recursos, abordando a metodologia utilizada na implementação do ambiente virtual, a padronização convencional e as relações métricas entre os cenários real e virtual.

Por fim, como sugestão de trabalhos futuros e melhorias para o sistema ITV podem ser citadas a possibilidade de utilização de estereoscopia, criação de modelos otimizados para aplicações *web*, simulações envolvendo cálculos físicos e adaptação dos mecanismos de interação para viabilizar o uso de *joysticks*.

Referências Bibliográficas

- [1] ALVES DE SOUSA, Marcos Paulo. **Uma nova abordagem de treinamento de manutenção em uma unidade hidrogeradora elétrica, utilizando tecnologias de realidade virtual desktop**. 2009. 124 f. Tese (Doutorado em Engenharia Elétrica) – Instituto de Tecnologia, Universidade Federal do Pará, Belém, 2009. p.112.
- [2] DALGARNO, Barney J., **Characteristics of 3D environments and potential contributions to spatial learning**. 2004. 453 f. Tese (Doutorado em Engenharia da Computação). Faculty of Education. University of Wollongong, 2004. pg. 206.
- [3] WALLER, David; MILLER, Jon. A desktop virtual environment trainer provides superior retention of a spatial assembly skill. In: ACM SIGGHI. 1998. Washington. **Anais...** Washington: Department of Psychology, 1998. p.1-3.
- [4] REID, Denise. **Benefits of a virtual play rehabilitation environment for children with cerebral palsy on perceptions of self- efficacy: a pilot study**. Toronto: University of Toronto. Department of Rehabilitation Science and Department of occupational Therapy, v. 5, n. 3, p. 141- 148, ago. 2002.
- [5] G. Ahlberg; T. Heikkinen; L. Iselius; C. Leijonmarck; J. Rutqvist; D. Arvidsson. Does training in a virtual reality simulator improve surgical performance. **Surgical Endoscopy and Other Interventional Techniques**. Nova Iorque. p.127, nov. 2001.
- [6] LOPES, Andrey; NUNES, Marcus V. A.; VILHENA, Paulo R. M.; RIBEIRO, Manoel. Utilizando técnicas de realidade Virtual para o desenvolvimento de programa computacional que objetiva a representação de uma unidade hidrogeradora integrada em um sistema elétrico. In: CONGRESSO BRASILEIRO DE AUTOMÁTICA. 2006. **Anais...** Belém: Universidade Federal do Pará. 2006. p. 1-6.
- [7] SHERMAN, William R. **Understanding Virtual Reality: Interface, application and design**. Califórnia: Elsevier science, 2003. ISBN 1-55860-353-0. p. 8, 9 e 11.

- [8] STRICKLAND, JONATHAN. **How Virtual Reality Gear Works**. Geórgia: HowStuffWorks, 2007. Disponível em: <<http://electronics.howstuffworks.com/gadgets/other-gadgets/VR-gear.htm>>. Acesso em: jun. 2010.
- [9] CYBERGLOOVE SYSTEMS LLC. **Cyber Gloove II**. Califórnia: CyberGlove Systems LLC, 2010. Disponível em: <<http://www.cybergloovesystems.com/sites/default/files/pg6.jpg>>. Acesso em: jun. 2010.
- [10] LOGITECH ASIA PACIFIC LTD. **Greater control with realistic shifter and foot pedals — designed by the pros at MOMO® Design**. Hong Kong: Logitech Asia Pacific Ltd. Disponível em: <<http://www.logitech.com/en-my/gaming/wheels/devices/320>>. Acesso em: jun. 2010.
- [11] GOOGLE. **O Google Earth**. São Paulo: Google, 2010. Disponível em: <<http://earth.google.com/intl/pt-BR/>>. Acesso em: jun. 2010.
- [12] UNITY TECHNOLOGIES. **Unity gallery**. São Francisco: Unity Technologies, 2010. Disponível em: <<http://unity3d.com/unity/>>. Acesso em: jun. 2010.
- [13] UNITY TECHNOLOGIES. **Room of Shadows**. São Francisco: Unity Technologies, 2010. Disponível em: <<http://unity3d.com/gallery/live-demos/index.html#shadows>>. Acesso em: jun. 2010.
- [14] RAHMAN, A. Abdul; PILOUK, M. **Spatial Data Modelling for 3d GIS**. Londres: Springer- Verlag Berlim Heidelberg New York. 2008. p. 30, 32, 41. ISBN 978-3-540-74166-4.
- [15] TAKE-TWO INTERACTIVE SOFTWARE. **Rock Star Game: GRAND THEFT AUTO SAN ANDREAS**. Nova Iorque: Take-Two Interactive Software 1998. Disponível em: <<http://www.rockstargames.com/sanandreas/>>. Acesso em: 21 jun. 2010.
- [16] ARGOSY PUBLISHING INC. **Visible Body®: 3D Medical animation & illustration**. USA: Argosy Publishing Inc., 2007. Disponível em: <http://www.visiblebody.com/demos/head/head_region_demo.html>. Acesso em: jun. 2010.

- [17] PIXAR ANIMATION STUDIOS. **Film Monsters, Inc: Behind the scenes**. Califórnia: Pixar Animation Studios, 1986. Disponível em: <http://www.pixar.com/featurefilms/inc/images/behind_copy.gif>. Acesso em: jun. 2010.
- [18] PIXAR ANIMATION STUDIOS. **Image Film Monsters, Inc**. Califórnia: Pixar Animation Studios, 1986. Disponível em: <<http://www.pixar.com/featurefilms/inc/images/monsters-front.jpg>>. Acesso em: jun. 2010.
- [19] ATWOOD, Jason. **Wankel Rotary engine cad conversion**. Geórgia: Georgia Institute of Technology, 2007. Disponível em: <<http://tir38.com/archives/73>>. Acesso em: jul. 2010.
- [20] WUSTEFELD, Tobias. **3D Pixel/ Voxel**. Hamburg, 2010. Disponível em: <<http://www.bilderzucht.de/blog/wp-content/uploads/2010/05/A-Voxel-Resolution.jpg>> Acesso em: jul. 2010.
- [21] WERNICK, Miles N.; ARSVOLD, John N. **Emission Tomography: The fundamentals of pet and spect**. Califórnia: Elsevier, 2004. ISBN 0-12-744482-3. p. 553.
- [22] LENGYEL, Eric. **Mathematics for 3D Game Programming and Computer Graphics**. 2.ed. Massachusetts: Charles River Media, 2004. ISBN 1-58450-277-0. p. 273, 453.
- [23] WIKIMEDIA COMMONS. **Non-uniform rational B-spline**. Califórnia: Wikimedia Foundation, 2010. Disponível em: <<http://en.wikipedia.org/wiki/NURBS>>. Acesso em: ago. 2010.
- [24] KLAWONN, Frank. **Introduction to Computer Graphics: Using Java 2D and 3D**. 1. ed. Londres: Springer - Verlag London Limited, 2008. p. 7, 8, 150. ISBN150.978-1-84628-847-0.
- [25] GUTHE, Michael; BALÁZS, Ákos; KLEIN, Reinhard. GPU-based trimming and tessellation of NURBS and T-Spline surfaces, In: ACM Siggraph. **Anais...**University Bonn: Institute of Computer Science II. p.1.

[26] RUSSO, Mário. **Poligonal Modeling; Basic and Advanced Techniques**. Texas: Wordware Publishing, 2006. ISBN 978-1-59822-007-0. p.IX, X.

[27] WIKIMEDIA COMMONS. **Polygon mesh**. Califórnia: Wikimedia Foundation, 2010. Disponível em: <http://en.wikipedia.org/wiki/Polygon_mesh>. Acesso em: ago. 2010.

[28] DAVISON, Andrew. **Killer Game Programming in Java**. 1. ed. Califórnia: O'Reilly Media Inc, 2005. ISBN 0-596-00730-2. p. 565.

[29] OGRE WIKI. **Particle Accelerator**. Ogre 3D, 2006. Disponível em: <<http://www.ogre3d.org/tikiwiki/Particle+Accelerator>>. Acesso em: ago. 2010.

[30] HO, Yo-Sung; KIM, Hyoung Joong. **Advances in Multimedia Information Processing - PCM 2005**. v.1. Department of Control and Instrumentation Engineering. Korea: Springer, 2005. ISBN 978-3-540-30027-4. . p. 74.

[31] BURDEA, Grigore C.; COIFFET, Philippe. **Virtual reality technology**. 2. ed. New Jersey: John Wiley & Sons Inc, 2003. ISBN 0-471-36089-9. p. 164-165.

[32] BRUNETT, Guido; HAMMANN, Bernd; LINSEN, Lars; MULLER, Heinrich. **Geometric modeling for scientific visualization**. Nova Iorque: Springer-Verlag Berlin Heidelberg, 2000. ISBN 3-540-40116-4. p. 207.

[33] MUDORCK, Kelly L. **3ds Max® 9 Bible**. Indianópolis: Wiley publishing Inc, 2007. ISBN 978-0-470-10089-9. p. 505.

[34] JUNKER, Gregory. **Pro Ogre 3D Programming**. Nova Iorque: Springer-Verlag New York, 2006. ISBN 978-1-559059-710-1. p. 39.

[35] AUTODESK, INC. **Autodesk 3ds Max Products**. Califórnia: Autodesk, Inc., 2010. Disponível em: <<http://usa.autodesk.com/adsk/servlet/pc/index?siteID=123112&id=13567410>>. Acesso em: ago. 2010.

[36] BLENDER FOUNDATION. Site Blender. Amsterdã: Blender Foundation, 2010. Disponível em: <<http://www.blender.org/>>. Acesso em: ago. 2010.

- [37] ADOBE SYSTEMS INCORPORATED. **Adobe® Photoshop® CS5**. Nova York: Adobe Systems Incorporated, 2010. Disponível em: <<http://www.adobe.com/br/products/photoshop/photoshop/>>. Acesso em: ago. 2010.
- [38] GNOME FOUNDATION. **Gimp 2.6**. Gnome foundation: Massachusetts, 1997. Disponível em: <www.gimp.org>. Acesso em: ago. 2010.
- [39] DEITEL, Paul; DEITEL, Harvey M. **C: How to Program**. 6. ed. ISBN 978-0-13-612356-9.
- [40] NIELSEN, Jakob. **Usability Engineering**. 1 ed. Califórnia, 1993. ISBN: 1-12-518406-9. p. IX
- [41] NASCIMENTO, Messias J. A. do; FILHO, Manoel Ribeiro; BARATA, Pebertli N. A.; JUNIOR, Alcides R. da S. Pamplona. Instruções Técnicas de Manutenção e Operação Virtuais de Uma Unidade Geradora Hidráulica. In: SEMINÁRIO NACIONAL DE PRODUÇÃO E TRANSMISSÃO DE ENERGIA ELÉTRICA. 2009. **Anais...** Recife: Universidade Federal do Pará. 2009.
- [42] LANDSAT, **Free Global Orthorectified Landsat Data via HTTP**. 2010. Disponível em: <<http://landsat.org/ortho/index.php>>. Acesso em: nov. 2009