

Desenvolvimento de Jogos Ultra-Alta Definição (UHD-4K)

Maria Amelia Eliseo*

Luciano Silva**

Universidade Presbiteriana Mackenzie, Faculdade de Computação e Informática, Brasil



Figura 1: Imagens do jogo Fifa 16, produzido pela EA Sports, que utiliza vídeo UHD em sua configuração para manter o realismo da simulação do jogo de futebol. Fonte: <http://www.techradar.com>.

ABSTRACT

The UHD-4K technology refers to video signals with the typical resolution of 2160×3840 pixels. These signals need support high performance and quality requirements, both in hardware and software. Recently, with the advent of new GPU technologies, projectors, monitors and graphics libraries, there was a growing interest in digital games for UHD-4K environments. Within this context, the aim of this work is to present the fundamental bases of UHD-4K video signals and transport them to game applications, going from the hardware specifications to the more specific requirements as rendering aware resolution and geometric tessellations.

Keywords: UHD-4K, HD resolution, High Resolution Systems, Digital Games.

RESUMO

A tecnologia UHD-4K refere-se a sinais de vídeo com a resolução típica de 2160×3840 pixels. Estes sinais necessitam de suporte com altos requisitos de desempenho e qualidade, tanto em hardware quanto em software. Recentemente, com o advento de novas tecnologias de GPU, projetores, monitores e bibliotecas gráficas, houve um crescimento do interesse de jogos digitais para o ambiente UHD-4K. Dentro deste contexto, o objetivo deste texto é apresentar as bases de funcionamento dos sinais de vídeo UHD-4K, assim como transportá-los para aplicações de jogos digital, passando desde as especificações de hardware até os requisitos mais específicos como renderização ciente de resolução e tesselações geométricas.

Palavras-chave: UHD-4K, Vídeo de Alta Resolução, Sistemas de Alta Resolução, Jogos Digitais.

1 INTRODUÇÃO

Com o advento de plataformas gráficas e software de alta performance, os jogos digitais começam a especificar requisitos

cada vez mais exigentes [1]. Um destes requisitos refere-se a resoluções gráficas de alta qualidade como a Ultra-Alta Definição ou *Ultra-High Definition* (UHD), que permitem trabalhar com áreas maiores tanto de projeção quanto de visualização.

O formato de vídeo UHD vem atraindo o interesse na área do desenvolvimento de jogos, devido a sua capacidade para apresentar imagens de ótima qualidade. O Jogo Fifa 16, mostrado na Figura 1, uma simulação de jogo de futebol produzido pela EA Sports para Microsoft Windows, PlayStation 3, PlayStation 4, Xbox 360, Xbox One, Android e iOS é um exemplo que utiliza UHD em sua configuração. Este formato de vídeo, define parâmetros avançados em múltiplos aspectos de um sinal de vídeo. O vídeo UHD suporta não só resoluções espaciais maiores (imagens com 3.840×2.160 px e 7.680×4.320 px) e taxas de quadros mais altas (até 120 Hz) do que HD, mas também profundidades de bits mais elevada da amostragem (até 12 bits para suporte de alta faixa dinâmica) e uma maior gama de cores que permite a renderização de cores mais nítidas [2].

Dentro de UHD, existem duas categorias básicas: UHD-4K (3.840×2.160 px) e o UHD-8K (7.680×4.320 px). Dentre estas categorias, o UHD-4K é o que possui maior suporte tanto em termos de hardware quanto software. Além disto, por ser mais antigo que o UHD-8K, o UHD-4K já possui uma gama maior de aplicações em jogos [3]. E ainda, o UHD-8K gera um custo significativo alto, demandando uma infraestrutura com maior capacidade. Mesmo assim, ainda existem diversas oportunidades tanto em pesquisa quanto desenvolvimento de produtos para o ambiente UHD-4K, passando desde a especificação e implementação de hardware gráfico dedicado até algoritmos e técnicas específicas para processos gráficos.

Dentro deste contexto, este capítulo apresenta as bases dos sinais de vídeo UHD-4K, focando suas aplicações especificamente no campo dos jogos digitais. Além da apresentação das plataformas de hardware e software, são apresentados os requisitos específicos dos ambientes UHD-4K quanto à diversidade de modos de vídeo, necessidades especiais de texturização, esquemas de filtragem, além de alguns detalhes em termos de processamento geométrico.

O texto está organizado da seguinte forma:

*e-mail: mamelia@mackenzie.br

**e-mail: luciano.silva@mackenzie.br

- a primeira seção apresenta os conceitos básicos de UHD, realizando a divisão clássica UHD-4K e UHD-8K e comparando-as com as resoluções tradicionais;
- em seguida, são apresentados os principais componentes de hardware para suportar UHD-4K como GPUs, projetores e monitores;
- seguindo, são apresentadas as principais bibliotecas que possuem suporte a UHD-4K como OpenGL, DirectX e SDL, assim como exemplos de sistemas mais interativos de desenvolvimento como Unity3D e Quest3D;
- finalmente, são apresentados os principais detalhes de desenvolvimento de aplicações em jogos com UHD-4K, focando-se nos detalhes específicos desta resolução.

O texto é finalizado com um conjunto de referências bibliográficas de suporte inicial aos interessados tanto em projeto quanto em desenvolvimento de jogos para ambientes UHD-4K.

Os autores esperam que este pequeno texto inicial possa abrir novas oportunidades de pesquisa e desenvolvimento de jogos com requisitos bastante exigentes tanto em relação à qualidade gráfica quanto em desempenho computacional.

2 FUNDAMENTOS DE ULTRA-ALTA DEFINIÇÃO

Este capítulo apresenta os conceitos básicos sobre vídeos de UHD, mostrando a qualidade dos sinais deste formato e mostra algumas aplicações na área de jogos digitais.

2.1 Conceitos Básicos de Ultra-Alta Definição

O termo Ultra-Alta Definição (do inglês, *Ultra High Definition* – UHD) [4] é composto, atualmente, por dois grandes formatos de vídeo digital:

- UHD-4K: criado pela SONY em 2003, trabalha com uma resolução de 2160×3840 pixels (~8.3 megapixels);
- UHD-8K: criado pela SONY em parceria com a SHARP, define uma resolução de 4320×7680 pixels (~32 megapixels).

A Figura 2 ilustra, em escala, os tamanhos comparativos dos formatos UHD-4K e UHD-8K entre si e com outros formatos bem conhecidos em mercado: FHD (*Full High Definition*), ou simplesmente HD, e SD (*Standard Definition*).

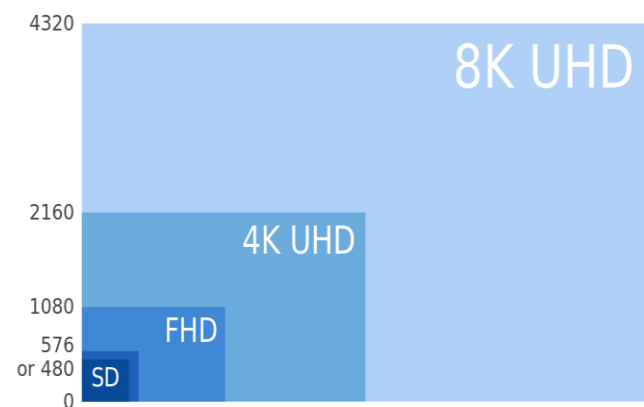


Figura 2: Quadro comparativo entre os formatos SD, FHD, UHD-4K e UHD-8K. Fonte: www.sony.com

Tomando-se como referência a densidade de pixels do formato FHD (~2.1 megapixels, conhecido também pelo nome 2K), o formato UHD-4K possui, aproximadamente, o quádruplo desta densidade (~8.3 megapixels) e, por sua vez, o UHD-8K (~32 megapixels) possui 8 vezes a densidade de um UHD-4K. A tabela 1 mostra uma comparação entre os formatos de vídeo FHD e UHD. Além de uma definição espacial superior ao do FHD, o formato UHD possui uma gama de cores mais ampla e com mais

profundidade, ou seja, as cores são mais nítidas. Os sinais de 10 e 12 bits são capazes de representar 1,07 bilhões e 6,87 bilhões de cores, respectivamente [2].

Tabela 1. Comparação entre os formatos de vídeo de alta definição (HD) e ultra-alta definição (UHD). Adaptada de [2].

		FHD	UHD
Espaço	pixels	1920 x 1080	3840 x 2160 (4K) 7680 x 4320 (8K)
Tempo	Taxa de Quadros	60, 50, 30, 25, 24	120, 60, 50, 30, 25, 24
	Scan	Progressivo, Entrelaçado	Progressivo
Cores Primárias	Vermelho	(0.640, 0.300)	(0.708, 0.292)
	Verde	(0.150, 0.330)	(0.170, 0.797)
	Azul	(0.600, 0.060)	(0.131, 0.046)
Codificação		8 e 10 bit	10 e 12 bit

Este capítulo, em particular, tem interesse pelo formato UHD-4K, devido ao aumento de disponibilidade tanto de hardware quanto software para desenvolvimento de aplicações para jogos digitais.

2.2 Sinais de Vídeo em UHD-4K

A qualidade dos sinais em UHD-4K é bastante perceptível quando comparada àquela do FHD. A Figura 3 ilustra, para uma pequena região de pixels, a diferença de qualidades entre os dois sinais:

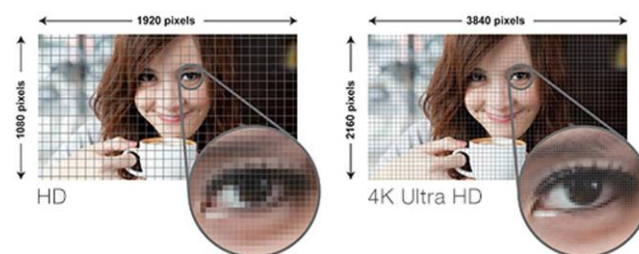


Figura 3: Comparação de qualidade de sinal entre FHD(HD) e UHD-4K. Fonte: www.sony.com

Se for considerada uma mesma área para os formatos FHD e UHD-4K, neste último a quantidade de pixels/polegada (*ppi-pixels per inch*) deverá ser maior que o FHD. Este acúmulo de pixels por polegada é um dos fatores que contribui para que uma imagem seja considerada com maior qualidade no UHD-4K do que em FHD. A Tabela 2 mostra um comparativo entre as medidas de pixels/polegada mais comuns para displays UHD-4K e FHD [4].

No formato UHD-4K encontra-se, aproximadamente, o dobro de pixels/polegada quando se compara com o formato FHD.

Um desafio para a reprodução de quantidade de dados transportados pelo sinal de vídeo UHD em jogos digitais, é reduzir significativamente esta quantidade de dados sem comprometer a qualidade do vídeo. Para isso são utilizados algoritmos de compressão de vídeo, como o H.264/AVC (*Advanced Video Coding*) e mais recentemente o padrão *High Efficiency Video Coding* (HEVC), finalizado em 2013 [2] [5].

Tabela 2: Quantidade de pixels/polegada em displays UHD-4K e FHD

	4K UHD / 3840x2160	Full HD / 1920x1080
110"	40.05	20.03
84"	52.54	26.23
75"	58.74	29.37
65"	67.78	33.89
64.5"	68.31	34.15
60"	73.43	36.72
58"	78.96	37.98
56.2"	78.4	39.2
55"	80.11	40.05
50"	88.12	44.06
46"	95.78	47.89
40"	110.15	55.07
39"	112.97	56.48
32"	137.68	68.84

2.3 Aplicações de UHD-4K em Jogos Digitais

A principal aplicação de UHD-4K em jogos digitais está concentrada em displays de alta resolução [6] como, por exemplo, em cavernas para imersão. Um destes exemplos é a caverna VisCube C4-4K (Figura 4), que possui quatro paredes de resolução 4K (4096x2160):

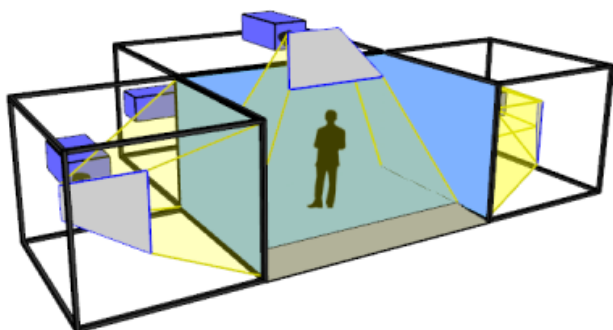


Figura 4: Caverna VisCube C4-4K para Realidade Virtual Imersiva. Fonte: www.visbox.com

Sistemas 4K imersivos têm sido usados com bastante sucesso na área de simuladores, como o exemplo de simulação de automóveis desenvolvido pela Renault, mostrado na Figura 5:

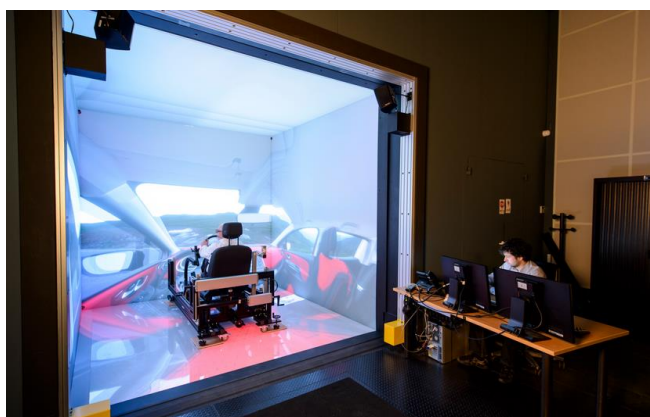


Figura 5: Ambiente imersivo 4K para simulação de direção de automóveis. Fonte: www.renault.com

Numa linha de menor custo, algumas aplicações de UHD-4K estão sendo desenvolvidas para monitores e aparelhos de TV 4K, como no exemplo do monitor HP ZVR na Figura 6:



Figura 6: Monitor 4K HP ZVR. Fonte: www.hp.com

O HP ZVR, além da resolução 4K, ainda possui um sistema de projeção estéreo de vídeo e um dispositivo (caneta) para manipulação 3D dos objetos em tela. Com a queda do custo dos monitores 4K, vários sistemas de jogos 3D baseados em Realidade Virtual puderam ser desenvolvidos. A Figura 7 mostra um exemplo do jogo 3D *Splinter Cell* num monitor 4K de baixo custo:



Figura 7: Jogo 3D em exibição num monitor 4K. Fonte: www.asus.com

3 HARDWARE PARA JOGOS DIGITAIS EM UHD-4K

O hardware necessário para suportar aplicações de jogos digitais em UHD-4K concentra-se em três componentes específicos:

- Unidades de Processamento Gráfico (GPU) com capacidade de processamento e saídas em UHD-4K;
- Projetores com entradas em UHD-4K;
- Monitores com entradas em UHD-4K.

Há diversos modelos de GPUs com capacidade de processamento e saídas em UHD-4K no mercado [7]. Um exemplo delas, mostrada na Figura 8, é a GPU Nvidia GeForce GTX 980:



Figura 8: GPU GeForce GTX 980. Fonte: www.nvidia.com

Esta GPU possui quatro saídas HDMI, que permitem o controle de até quatro monitores Full HD simultâneos com a tecnologia Nvidia SurroundTM. A configuração mostrada na Figura 9, por exemplo, utiliza três monitores principais de visualização e um de controle [8]:



Figura 9: Quatro monitores FHD controlados por uma GPU Nvidia GTX 980. Fonte: www.nvidia.com

Com a tecnologia DSR (*Dynamic Super Resolution*), um sinal de vídeo de FHD pode ser escalado pela GeForce 980 até a resolução UHD-4K e, vice-versa. O DSR inclui filtros em hardware tanto para *upscaling* (FHD → UHD-4K) quanto para *downscaling* (UHD-4K → FHD).

O segundo componente importante refere-se aos projetores UHD-4K, bastante utilizados em jogos do gênero simulador. Assim como as GPUs, também existem diversos modelos de projetores com esta resolução disponíveis no mercado. Um exemplo destes projetores é o Galaxy 4K-32, mostrado na Figura 10:



Figura 10: Projetor 4K com suporte a estéreo 3D. Fonte: www.barco.com

Além do suporte à projeção em UHD-4K, ele também suporta sinais em 3D estéreo. Para uma arquitetura de caverna digital em UHD-4K, pode-se utilizar quatro GPU GeForce GTX 980 (conforme Figura 11), cada uma controlando a saída para um projetor Galaxy 4K-32.



Figura 11: Possível configuração de GPUs para suporte a quatro projetores em uma caverna digital. Fonte: www.nvidia.com

Uma alternativa aos projetores UHD-4K, que ainda possuem um custo elevado em mercado, são os monitores UHD-4K. Existem diversas opções destes monitores e eles são divididos em duas grandes categorias, conforme mostrado na Figura 12:

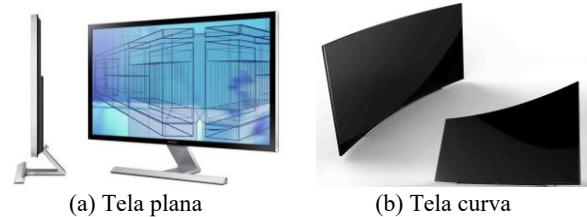


Figura 12: Categorias de monitores UHD-4K. Fonte: www.samsung.com

Todo o trabalho de geração de sinais UHD-4K para este tipo de monitor é realizado pelas saídas das GPUs. Conforme já discutido anteriormente, uma mesma GPU pode controlar vários monitores simultaneamente.

Alguns monitores de tela plana, como o HP ZVR citado anteriormente, ainda incluem um sistema de rastreamento [9] para interação com os objetos virtuais (Figura 13).



Figura 13: Monitor HP ZVR com sistema de rastreamento para interação. Fonte: www.hp.com

4 SOFTWARE PARA JOGOS EM UHD-4K

Grande parte das bibliotecas e *frameworks* conhecidos para desenvolvimento de aplicações RV (OpenGL, DirectX, SDL, dentre outros) já possuem suporte para UHD-4K. Na realidade, em muitas situações, o processamento UHD-4K é muito mais uma tarefa de hardware do que software. A seguir, serão detalhados alguns exemplos de códigos nas principais bibliotecas acima para trabalho com UHD-4K.

4.1 UHD-4K em OpenGL

Em OpenGL, a especificação do modo UHD-4K começa logo na inicialização do modo de vídeo, conforme mostrado no exemplo abaixo:

```
1 int main (int argc, char **argv) {
2     glutInit (&argc,argv);
3     glutInitDisplayMode (GLUT_DOUBLE | GLUT_DEPTH);
4     glutGameModeString( "3840x2160:32@60" );
5     ...
6 }
```

Explicitamente, na linha 4 do código acima, a resolução UHD-4K é utilizada, juntamente com a profundidade de cores de 32 bits e taxa de atualização de 60fps.

4.2 UHD-4K em DirectX

Em DirectX, a especificação do modo UHD-4K se inicia no momento em que se define a janela de renderização, conforme mostrado no código abaixo:

```
1 #define SCREEN_WIDTH 3840
2 #define SCREEN_HEIGHT 2160
3 hWnd = CreateWindowEx( NULL,
                        L"WindowClass",
                        L"Our Direct3D Program",
                        WS_OVERLAPPEDWINDOW,
                        300, 300,
                        SCREEN_WIDTH, SCREEN_HEIGHT,
                        NULL,
                        NULL,
                        hInstance,
                        NULL);
```

A especificação da resolução UHD-4K ocorre na linha 3, no momento da criação da janela principal de renderização.

4.3 UHD-4K em SDL

Em SDL, a especificação do modo UHD-4K se inicia no momento em que se define a superfície de renderização, conforme mostrado no código abaixo:

```
1 #include <iostream>
2 #include "SDL.h"
3
4 ...
5 SDL_Init(SDL_INIT_VIDEO);
6
7 // Set the video mode
8 SDL_Surface* myVideoSurface = SDL_SetVideoMode(
    3840, 2160, 16, SDL_DOUBLEBUF|SDL_FULLSCREEN);
9 if (myVideoSurface != NULL) {
10     std::cout << "The current video surface bits per
        pixel is " <<
        (int)myVideoSurface->format->
        BitsPerPixel << std::endl;
11 }
12 else {
13     std::cerr << "Video initialization failed: "
        << SDL_GetError()
        << std::endl;
16 }
17
18 SDL_Quit();
```

A especificação do modo UHD-4K no código acima é realizada na linha 8, onde se pode observar claramente a resolução 2160x3840 na seleção do modo de vídeo.

Não é difícil observar que, tanto em SDL, quanto DirectX e OpenGL, o procedimento de inicialização do modo UHD-4K é muito semelhante: utiliza-se o comprimento e a altura da resolução no momento de criação da área de renderização.

Embora os códigos possam passar a falsa ideia de que todo o restante da programação continua como nas resoluções mais baixas, existe uma série de boas práticas em UHD-4K especificamente relacionadas à garantia da qualidade gráfica, como texturas em altíssima resolução ou filtragem dinâmica, assim como àquelas relacionadas à questão de alto desempenho, como uso de CUDA ou OpenCL. Isto será detalhado em uma das seções posteriores.

Além das bibliotecas mostradas anteriormente, existem ambientes de mais alto nível como os ambientes Unity3D ou Quest3D, que também suportam a resolução exigida em UHD-4K.

Estes ambientes permitem a prototipação mais rápida de aplicações para RV, uma vez que já permitem uma visualização imediata do ambiente virtual que está sendo planejado e implementado. A seguir, serão mostrados exemplos destes dois ambientes.

4.4 Unity3D 4K

O ambiente Unity3D já disponibiliza suporte para resolução UHD-4K, que já pode ser observada diretamente na janela de projeto do ambiente virtual, conforme mostrado no exemplo da Figura 14, abaixo:

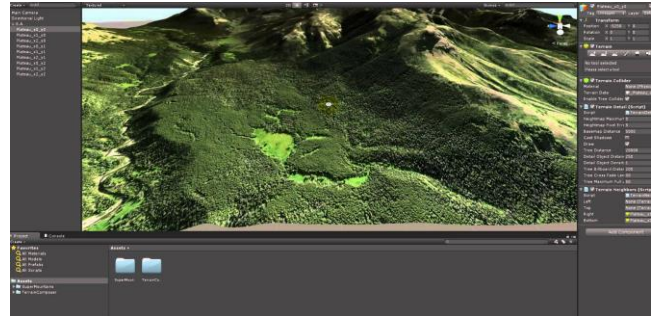


Figura 14: Ambiente Unity3D, mostrando modelagem de terreno virtual em UHD-4K. Fonte: www.unity3d.com

Além dos próprios recursos de integração de artefatos 3D, a Unity3D também possui integração direta com diversos dispositivos de interação para aplicações em RV.

4.5 Quest3D 4K

O ambiente Quest3D é um outro exemplo de plataforma com capacidades 3D para UHD-4K. A Figura 15 ilustra um exemplo desta plataforma.

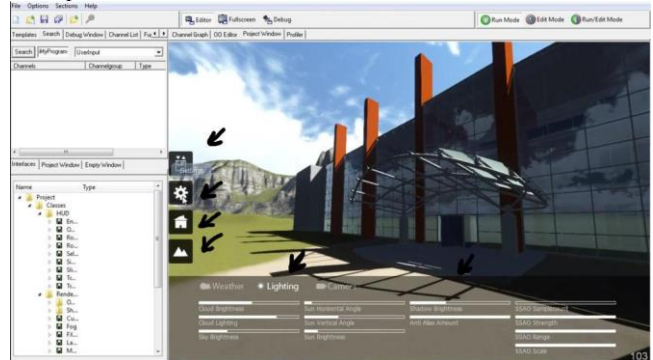


Figura 15: Ambiente Quest3D, mostrando modelagem de construção virtual em UHD-4K. Fonte: www.quest3d.com

Uma das grandes funcionalidades do Quest3D é a capacidade de programação visual, utilizada para construção de um grafo de cena.

5 DESENVOLVIMENTO DE JOGOS DIGITAIS EM UHD-4K

O desenvolvimento de jogos digitais para UHD-4K, tendo como suporte o hardware e software detalhado nas seções anteriores, utiliza uma grande parte do processo de desenvolvimento de aplicações em baixa resolução. Porém, alguns detalhes são específicos da resolução UHD-4K, que serão detalhados a seguir.

5.1 Renderização Ciente de Resolução

Para trabalhar com uma aplicação portátil entre os diversos tipos de resolução e modos de vídeo diferentes, é necessário listar os diversos dispositivos gráficos disponíveis, assim como seus modos de vídeo, conforme mostrado nos exemplos abaixo (DirectX):

```

BOOL EnumDisplaySettings(          HRESULT GetDisplayModeList(
    _In_ LPCTSTR lpszDeviceName,    [in]     DXGI_FORMAT EnumFormat,
    _In_ DWORD iModeNum,           [in]     UINT Flags,
    _Out_ DEVMODE *lpDevMode       [in, out]  UINT *pNumModes,
);                                 [out]    DXGI_MODE_DESC *pDesc
                                   );

```

A primeira função enumera os diversos dispositivos disponíveis e, a segunda, os diversos modos de vídeo [10].

5.2 Texturas em Alta Resolução

Resoluções UHD-4K também exigem texturas de alta resolução. Normalmente, utilizam-se texturas de tamanho 4096x4096 para se evitar efeitos de distorção, principalmente nos processos de upscaling [11]. A Figura 16 ilustra a comparação entre texturas de baixa resolução (512x512, à esquerda) e alta resolução (4096x4096, à direita) em um ambiente UHD-4K:



Figura 16: Comparação entre textura de baixa resolução e alta resolução em UHD-4K. Fonte: www.nvidia.com

5.3 Filtragem Anisotrópica

A resolução UHD-4K permite a exibição de grande nível de detalhe. Assim, esquemas de filtragem que maximizem a diferenciação destes detalhes é de grande importância. Um dos esquemas mais utilizados para filtragem em UHD-4K é a filtragem anisotrópica [11], cujo efeito é mostrado na Figura 17:



Figura 17: Filtragem gaussiana (esquerda) e anisotrópica (direita). Fonte: www.nvidia.com

Na parte esquerda da imagem, onde foi utilizada a filtragem gaussiana, é possível perceber a ocorrência de *blurring* (borramento). Por outro lado, à direita, os efeitos provenientes de *aliasing* são muito atenuados com a filtragem anisotrópica e há uma exacerbação dos detalhes.

5.4 Tesselação

A resolução UHD-4K também permite que se trabalhe com objetos com malhas poligonais muito grandes. Assim, a falta de uso

técnicas adequadas de tesselação [11], alguns efeitos de baixa podem ocorrer, como o mostrado na Figura 18:

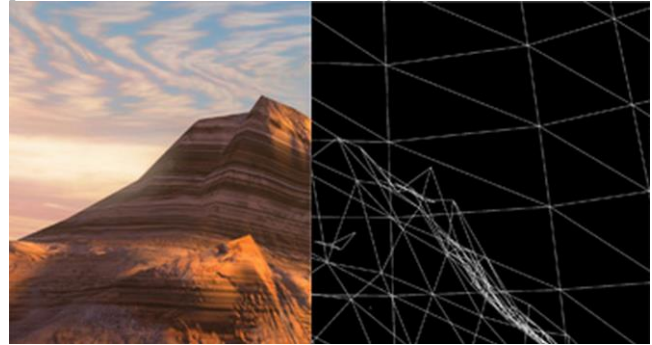


Figura 18: Objeto poligonal de baixa qualidade com baixa tesselação. Fonte: www.nvidia.com

Com a aplicação de um esquema de tesselação mais refinado, pode-se alcançar um nível maior de qualidade, como mostrado na Figura 19:

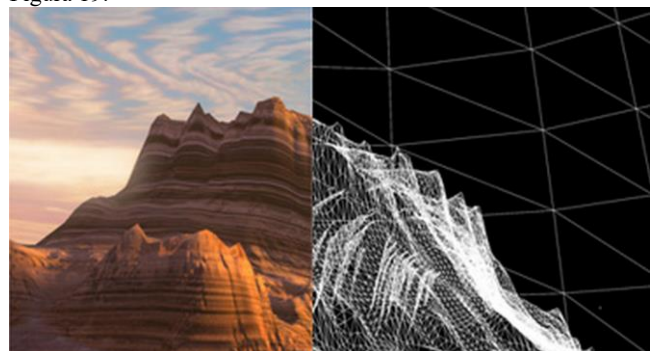


Figura 19: Objeto poligonal de alta qualidade com alta tesselação. Fonte: www.nvidia.com

Pode-se utilizar, ainda, esquemas de tesselação adaptativa, que permite regular quais partes do modelo necessitam maiores subdivisões do que outros.

REFERÊNCIAS

- [1] M. Mihelj, D. Novak and S. Begus. *Virtual Reality Technology and Applications*. New York: Springer, 2014.
- [2] Y. Ye, Y. He and X. Xiu. Manipulating Ultra-High Definition Video Traffic. In *IEEE MultiMedia*, vol. 22, no. 3, pp. 73-81, 2015.
- [3] C. Bates. *4K Resolution 99 Success Secrets: 99 Most Asked Questions on 4K Resolution*. New York: Emereo Publishing, 2014.
- [4] C. Poynton. *Digital Video and HD: Algorithms and Interfaces*. 2.ed. New York: Morgan Kaufmann, 2012.
- [5] W. Hamidouche, G. Cocherel, J. L. Feuvre, M. Raullet and O. Déforges. 4K real time video streaming with SHVC decoder and GPAC player. In: *Multimedia and Expo Workshops (ICMEW)*, 2014 IEEE International Conference on, 2014.
- [6] A. Kemeny. Photo-Realistic Digital Mock-Up Design Review in a Five 4Kx4K Sided CAVE. In: *GPU Technology Conference*, 1(1), pp. 30-15, 2014.
- [7] G. Barlas. *Multicore and GPU Programming: An Integrated Approach*. New York: Morgan Kaufmann, 2014.
- [8] F. Heide, D. Lanman, D. Reddy, J. Kautz, K. Pulli and D. Luebke. Cascaded Displays: Spatiotemporal superresolution using offset pixel layers. In: *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2014*, 33(4), pp. 40-51, Julho, 2014.
- [9] H. Jung, D. Kim and H. Kim. Implementation of UHD Interactive Media Art Work through MultiTouch Interface. In: *International*

Journal of Multimedia and Ubiquitous Engineering, 9(7), pp.87-96, 2014.

- [10] P. Varcholik. *Real-Time 3D Rendering with DirectX and HLSL: A Practical Guide to Graphics Programming*. New York: Addison-Wesley Professional, 2014.
- [11] W. Engel. *GPU Pro 6: Advanced Rendering Techniques*. New York: AK Peters, 2015.