

7. Conclusão e Trabalhos Futuros

7.1 Conclusões Finais

Neste trabalho, foram apresentados novos métodos para aceleração, otimização e gerenciamento do processo de renderização de cenas fotorealistas e de grande complexidade requeridas pela indústria de cinema e TV. O trabalho foi dividido em três módulos principais, que são resumidos a seguir.

O primeiro módulo apresenta uma metodologia de ganho de desempenho, otimizando os processos locais de renderização. Nesta metodologia, o renderizador é capaz de gerar estimativas ou aproveitar resultados de *frames* próximos para realizar um processo inovador de subdivisão de *pixels* para serem processados pelos núcleos de um computador, ou serem enviados para a renderização em *GPU*. Essa metodologia proporciona um ganho de, aproximadamente, 10% na renderização local, quando comparado com sistemas extremamente otimizados. Ainda, o trabalho mostra uma forma de especificação de materiais que permite ao renderizador a geração dinâmica de materiais em diversos formatos. Essa técnica inovadora permite ao renderizador que técnicas de renderização sejam alternadas sem a reespecificação dos materiais envolvidos.

Quando múltiplas técnicas são empregadas foi desenvolvida uma metodologia de composição de imagens que utiliza os canais de profundidade (canais que levam em consideração técnicas com *Bump Map* e *Displacement Map*). Assim, tendo como base o processo de composição que envolve artistas experientes para realizar a combinação

correta das imagens com os devidos pesos, foi apresentada uma forma automática de se realizar a combinação.

Ainda no primeiro módulo, técnicas de geração e aproveitamento de *caches* de irradiação são apresentadas.

No segundo módulo, o trabalho introduz uma técnica nova de se avaliar o processo de gerenciamento de *Render Farms*, processo este que pode ser estendido para outras computações distribuídas. Nesse processo inovador, os *frames* e os servidores são avaliados e as decisões de como serão atribuídas as atividades aos servidores passam por um processo de análise prévia do problema e distribuição compensada de cargas. Isso difere de outras metodologias pois é analisado o conteúdo trabalho a ser realizado. Além disso, o sistema apresenta uma forma de identificar e acompanhar o processo de distribuição de forma a entender fenômenos, como a extrapolação de tempo de renderização e servidores incapazes de realizar determinadas tarefas, isso feito de forma dinâmica. Adicionalmente, esse gerenciador é capaz de subdividir *frames* de forma equilibrada de carga de trabalho e reacomodar e alocar máquinas, sejam de grupos de servidores locais, como estações de trabalho ociosas, bem como servidores disponibilizados em *clusters* de nuvens públicas.

Dessa forma, o gerenciador é capaz de assumir e adequar todo o processo de renderização para atender três tipos de restrições: Tempo, Recurso e Custo (quando associado à nuvem), ajustando a quantidade de máquinas e até a técnica de renderização para atender à restrição. Além disso, o tratamento da nuvem é feito utilizando os recursos da mesma da forma como eles foram concebidos e não adaptando a solução para integrar com os sistemas.

No terceiro módulo são apresentados meios de aumentar a eficiência da renderização através do compartilhamento de informações entre os nós de processamento, disponibilizando dados em execução para acelerar o processo de outros nós no *cluster*/nuvem. Essa metodologia, chamada de renderização colaborativa, é extremamente

inovadora, visto que não existem trabalhos correlatos para renderização e é um campo ampo para melhorias. O ganho médio do uso foi de 2%, sendo que esse processo atua apenas na área gerencial do renderizador.

Além disso, são realizados diversos testes de desempenho, os quais mostram que a metodologia realmente melhora o processo distributivo e de processamento, apenas reduzindo as perdas e ociosidades dos servidores.

Cabe notar que a grande diferença entre cenas e processos desenvolvidos para renderizadores de tempo real e sistemas não voltados para o cinema e TV é a enorme diferença de escala dos elementos envolvidos. Assim, o processo de renderização de cenas reais envolve uma quantidade de materiais, luzes, texturas e geometrias muito superior ao apresentado em diversos trabalhos, de forma que as necessidades para atender a demanda envolvem processos que, normalmente, reduzem desempenho, como *streaming* (carregamento dinâmico da cena). No entanto, apesar de serem processos caros, eles são fundamentais para realizar a renderização, ou seja, podem existir ganhos muito maiores se não forem empregadas essas técnicas.

Adicionalmente, o presente trabalho foca na melhoria dos processos de forma a utilizar os recursos disponíveis para a renderização, reduzindo os desperdícios de processamento. Dessa forma, a presente tese cumpre o objetivo de criar uma metodologia de utilização do hardware e recursos de comunicação de forma muito mais eficiente, apesar de, em média, apresentar ganhos na ordem de 10% apenas. Entretanto, este valor médio foi obtido utilizando-se compiladores não otimizados ao invés de compiladores de alto desempenho (e.g. Intel compiler, cuja estimativa de ganho é maior do que 5% (Intel, 2015)). Além disso, deve-se avaliar que esse ganho de 10% reflete o ganho real na cadeia de produção, com cenas e problemas reais.

Ainda há muito campo para a experimentação de melhores soluções para a solução dos problemas de renderização e distribuição de

processamento nos *clusters* de computação gráfica. Porém, a contribuição apresentada por esse trabalho cria uma base a ser experimentada com outras técnicas (e.g. usando inteligência artificial) e em outras áreas da computação, não se restringindo apenas à renderização.

6.1.11. 7.2 Trabalhos Futuros

O processo de otimizar a renderização de cenas possui diversos elementos que podem ser melhorados. Esse trabalho mostrou diversos implementos que proporcionaram um avanço substancial no processo de renderização para cinema e TV. No entanto, pode-se elencar os seguintes tópicos para trabalhos futuros.

- Generalizar o processo de distribuição local e de *cluster* de forma a atender diversos tipos de problemas.
- O sistema de gerenciamento pode utilizar métodos mais avançados de inteligência artificial, como as redes neurais artificiais.
- A metodologia de decisão pode utilizar lógica nebulosa (fuzzy) para tomar decisões com mais suavidade, evitando trocas bruscas de estado e evitando chaveamentos de decisão (teste 10.1).
- Diversas heurísticas utilizadas no trabalho podem ser melhoradas e seus fatores recalculados. No entanto, esse é um processo que requer um volume grande de experimentações e testes.
- O processo geral apresentado nesse trabalho pode ser extrapolado para outras áreas da computação, como, por exemplo, a codificação de vídeos e processamento de imagens.
- O sistema de geração de Quadrees pode ser utilizado como base para a codificação de vídeo h.265, permitindo que ao final da renderização de um conjunto de blocos, os mesmos sejam facilmente encodados e gerem um bloco entre *key frames* do codificador h.265. Sendo possível combiná-los posteriormente.

- A avaliação de renderizadores mais abertos como o Arnold ou Corona, que permitem acessar dados de mais baixo nível e controlar alguns processos de distribuição. Assim, o gerenciador poderia ter melhor ganho de desempenho mesmo sem o uso do renderizador proposto nesta tese.
- Mais otimizações no renderizador desenvolvido, visto que nem todos os processos de otimização empregados pelos renderizadores comerciais foram desenvolvidos.
- Apesar do sistema ter sido usado em ambientes de produção, a necessidade de melhorias nas interfaces e métodos de integração com *softwares* comerciais é necessária.
- O processo de renderização colaborativa pode ser ainda muito ampliado e otimizado se forem utilizados os dados de *Irradiance Cache* entre os nós, permitindo a criação de um volume extraordinário de amostras, as quais podem melhorar o desempenho dos renderizados de forma extremamente alta.
- Pode-se melhorar o processo de alocação dinâmica na nuvem permitindo que o sistema escolha o hardware de acordo com a necessidade. Assim, nem sempre a máquina mais cara seria usada.
- Pode-se avaliar uma forma de renderização interativa que use os *frames* predecessores para subdividir a renderização.
- Um tópico não tão diretamente relacionado ao processo em si é a maneira de analisar os resultados de forma gráfica e paramétrica. Desta maneira, sugere-se uma investigação sobre como mostrar a evolução destes resultados em gráficos 3D, com eixos representando parâmetros, tais como número de luzes, número de triângulos, número de processadores e tipo de hardware. Infográficos dinâmicos também seriam interessantes de serem investigados.