

## Fluxo de trabalho para uma animação com *render* em tempo real

Igor Abrahão Arantes

Universidade FUMEC, Brasil

Bernardo Giorni Abijaude Bracarense

Universidade FUMEC, Brasil

Samuel Marinho Leidner

Universidade FUMEC, Brasil

Humberto de Campos Rezende

Universidade FUMEC, Brasil

João Victor Boechat Gomide

Universidade FUMEC, Brasil

### Abstract

*This paper presents the workflow for developing an animated short film, Fio, using real-time rendering. Real-time rendering is a recently commercially available alternative and has drastically reduced rendering time.*

*Rendering means graphically processing an image, or layers of images, so that it can be displayed after processing, immediately. It is a critical step in computer graphics and determines the quality of the final image and the time required for it to be available for display. Previously, rendering required many minutes for each image frame, with multiple and costly processors. Currently, one render per frame is achieved in less than a second, with an equivalent look and a smaller amount of processing.*

*To render in real time, the graphics engine Unreal Engine, a pioneer in this type of solution, was used. For this type of use, to render scenes, the software is free. Software of relevance in the creative industry were chosen as well, such as Autodesk Maya for animations and modeling, zBrush for sculptures, Substance Painter and Photoshop for composing textures, Audition for soundtracks, Premiere for editing. The engine itself has tools for lighting and composition. Other experimental alternatives were tested in the workflow, such as the use of hand-drawn textures with pastel chalks.*

*This paper will introduce each step of the workflow for producing an animated short film, with the preparation of the scene adjustments to be imported by the Unreal engine at the end. The first minute of this animation is accessible in <https://vimeo.com/503536788>.*

**Keywords:** Real time rendering, Animation short film, Animation workflow, UV texture, Render em tempo real

### Introdução

Este artigo apresenta o fluxo de trabalho para o desenvolvimento de um curta metragem de animação, com o nome *Fio*, utilizando o render em tempo real. O vídeo do primeiro minuto do curta metragem (*Fio* 2020) está em <https://vimeo.com/503536788>. Esta animação é o resultado do trabalho de conclusão de curso dos alunos Bernardo, Igor e Samuel, orientados

pelos professores João Victor e Humberto, realizada no laboratório-escola de Computação Gráfica da Universidade FUMEC.

O render em tempo real é uma alternativa disponível comercialmente há pouco tempo e tem permitido reduzir o tempo de renderização de forma drástica (Akenine-Möller et al 2018). Renderizar é um anglicismo, que significa processar graficamente uma imagem, ou camadas de imagens, para que ela possa ser exibida depois do processamento, imediatamente. É uma etapa crítica da computação gráfica, e determina a qualidade da imagem final e o tempo necessário para que ela esteja disponível para exibição. Até pouco mais de cinco anos atrás, o render necessariamente demandava um tempo de muitos minutos para cada quadro de imagem, com múltiplos e custosos processadores. Atualmente, se obtém um render por quadro em menos de um segundo, com um visual equivalente e uma quantidade menor de processamento. Para fazer o render em tempo real, foi utilizado o motor gráfico Unreal Engine, pioneiro neste tipo de solução, com imagens fotorealistas (Unreal Engine 2019).

Foram usados softwares de relevância na indústria criativa, como o Autodesk Maya para as animações e modelagem, o zBrush para esculturas, o Substance Painter e o Photoshop para composição de texturas, o Audition para trilha sonora, o Premiere para edição, bem como ferramentas disponibilizadas dentro do próprio motor gráfico para iluminação e composição. Outras alternativas experimentais foram testadas, como o uso de texturas desenhadas a mão com giz pastel.

Este artigo vai apresentar cada passo do fluxo de trabalho da animação *Fio*, com a preparação dos ajustes das cenas para o Unreal Engine no final.

O objetivo é realizar todo o processo de desenvolvimento de uma produção audiovisual de curta-metragem em 3D, assim como experimentar técnicas artísticas para intensificar a identidade do produto, e utilizar o render em tempo real. O intuito é solucionar um grande gargalo no tempo de produção, que é o tempo de renderização e explorar o potencial da ferramenta, que é uma opção alternativa que está entrando em ascendência de uso ao longo dos últimos anos.

## Trabalhos relacionados, usados como referência artística

O curta de animação (*A Boy and His Kite* 2015) é pioneiro na técnica utilizada. Ele foi exibido durante a SIGGRAPH 2015, que é um dos mais importantes congressos de computação gráfica em todo o mundo, recebendo o Prêmio de Melhor Gráfico em Tempo Real e Interatividade. O curta metragem foi dirigido por Gavin Moran e produzido e lançado pela Epic Games, que é a desenvolvedora do motor gráfico Unreal, pioneiro em render em tempo real de alta qualidade. O curta metragem foi usado como uma demo, para divulgar as novas propriedades da Unreal, com cenários realistas e apresentação de uma qualidade visual que rivaliza com as ferramentas de render mais utilizadas para o audiovisual, como na figura 1.



Figura 1: quadro do curta metragem de animação *A Boy and His Kite*, produzido pela Epic Games (Unreal Engine 2015)

Para a produção do curta metragem, foram utilizadas diversas fontes de referência. Uma delas é o longa metragem (*Up: Altas Aventuras* 2009), que inicialmente foi usado como um modelo para as formas e personalidade dos personagens e do mundo, mas também ajudou no aspecto de composição e no equilíbrio e desequilíbrio proposital de cada frame, para realçar o sentimento dos personagens. Já em (*Lost & Found* 2018), que é um curta em stop-motion independente, os personagens são feitos de crochê, e foi útil para reforçar a ideia dos animais serem feitos de lã em "*Fio*". Esses dois filmes foram a base da ideia do curta de animação.

Alguns conceitos visuais, tal como a iluminação e o aspecto do mundo exterior, especialmente da flora, surgiram apoiados no visual de (*Revolting Rhymes* 2016), que simplifica as formas e se assemelha a um mundo físico minimalista, aparentando um mundo esculpido a mão.

Nos elementos do interior da casa, foram analisadas cabanas e tavernas rústicas, principalmente em outros trabalhos estilizados. Uma das referências foi o curta metragem (*Hearthstone Animated Short: Win or Lose* 2019), tanto devido à presença de madeira e pedra usada nessas locações, quanto aos adereços.

Para a construção da trilha sonora, por um dos autores, Igor, foram analisados diversos filmes da Pixar. Alguns exemplos são (*Viva: a Vida é uma Festa* 2018); (*Monstros S.A.* 2001), (*Toy Story*, 1995) e o previamente citado (*Up: Altas Aventuras* 2009). Nestes

filmes são aplicadas técnicas de ênfase de temas para representar a relação de dois personagens, que são usadas em contraste, para destacar o sentimento na relação dos personagens.

Além de referências cinematográficas, foram analisadas imagens como inspiração nas decisões visuais ou para simplesmente criar ideias. Como as árvores da animação deveriam ser simples, o método usado foi baseado em análises de imagens de (*Vasek* 2020), que aparentam ser duplicadas, com pequenas variações nas folhagens, parecendo ser criadas em torno de uma forma circular.

Também foram analisadas plantas de cabanas pequenas, com dois quartos, para ajudar no planejamento da modelagem da casa.

## Pré-produção

### Sinopse e Elementos do Design de Produção

A linha do roteiro do filme é a seguinte: "Um velho e sua filha vivem costurando os animais de lã que habitam uma floresta, mas, infelizmente, nem tudo pode ser remendado." A partir do roteiro, foi realizado todo o processo de desenvolvimento do curta de animação, desde a pré-produção até a pós-produção e a finalização do curta metragem.

O design dos personagens principais teve como inspiração o filme (*O Menino e Mundo* 2013). Nesta animação, o minimalismo dos personagens passa a ideia de inocência e infantilidade, que são assuntos frequentes no decorrer da história. Isto contrasta com a seriedade e melancolia do final.

Um elemento que se manteve, desde o começo da produção, foi as cores atribuídas a cada personagem. O amarelo, que é uma cor saturada, viva e feliz, é atribuído à Filha, enquanto a cor atribuída ao Pai é um marrom envelhecido, que representa a idade e sabedoria, como na figura 2.



Figura 2 - concepções dos personagens de *Fio*

Os elementos do cenário, como árvores, folhagens, grama e o chão, tiveram que se alterar para cada estação do ano. Assim, as texturas mudam de acordo com a estação e são feitas com formas geométricas simples, como as folhagens das árvores em formato de círculo e a grama mais quadrangulada, como está na figura 3. Todas as texturas foram feitas à mão, com lápis pastéis, para ajudar no estilo único do curta.



Figura 3: formas das árvores e da grama do curta metragem

Na composição das sequências do curta metragem, são utilizadas técnicas clássicas de apresentação para cada nova cena. Assim, começa com um grande plano geral e vai se reduzindo as informações na tela, para revelar, aos poucos, a intenção da cena ou canalizar mais o foco em uma ação. Com esta intenção é trabalhado o posicionamento da câmera e dos personagens, para criar um equilíbrio ou desequilíbrio, de acordo com o sentimento que os personagens estão presenciando no momento. Os elementos de cena são compostos para que nada esteja presente em vão.

O figurino dos personagens segue as estações do ano. Para dar ênfase à passagem de tempo, foram criados dois figurinos para cada personagem, para acelerar o tempo gasto na produção, com cada figurino tendo diferentes texturas, para cada estação.

Devido à localização isolada da cabana, os principais materiais disponíveis nas redondezas são os mesmos que seriam usados na construção da moradia, que são a madeira e a pedra.

### Roteiro e storyboard

No início, a prioridade é a de concretizar o projeto, a partir da ideia inicial discutida entre os integrantes da equipe, na forma do argumento do curta. A partir disso, foi iniciado o processo de roteirização e da pesquisa de referências, para definir o estilo artístico do mundo onde a história se passa.

Após a conclusão do roteiro, foi feito o *storyboard*, a partir das cenas do roteiro. O *storyboard*, que está na figura 4, facilitou a observação de como cada cena se encaixa na estrutura geral do filme. A partir da primeira versão do *storyboard*, foram feitos vários ajustes na ordenação das cenas, assim como na adição e exclusão de outras, para que a história pudesse fluir melhor. A figura 4 mostra uma fotografia do início da elaboração do *storyboard*, com as primeiras tomadas, ainda no chão do laboratório. Em seguida, ele foi colado em um quadro de grandes dimensões, que comportava todos os desenhos, e onde era possível mover as tomadas utilizando fita adesiva.



Figura 4: fotografia das primeiras tomadas no *storyboard* de Fio

## Produção

### Modelagem

A primeira etapa de toda produção 3D é a modelagem. Esse processo consiste na criação de elementos, sejam eles personagens ou partes dos cenários, que irão compor a animação.

Essa etapa envolve a criação e manipulação de polígonos em um espaço tridimensional, levando em consideração apenas a topologia, que é a disposição dos polígonos em uma malha que representa um objeto 3D. Uma topologia otimizada é essencial para que facilite as próximas etapas dentro da produção, como na figura 5.

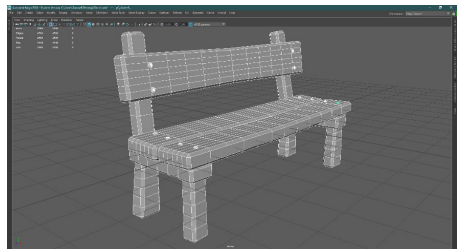


Figura 5: banco modelado no Maya, para ser exportado para a Unreal

Em produções cinematográficas, geralmente são utilizadas técnicas de subdivisão, que dividem a malha para criar uma superfície mais suave. Devem ser criados os *edge-loops*, que criam arestas em linha ao longo de todo o objeto. Para que a geometria consiga calcular os *edge-loops*, os polígonos devem ter quatro lados, o que é um desafio, dependendo da forma desejada e do nível de detalhe do objeto. Esta etapa é fundamental para que as ferramentas de render possam processar os polígonos e suas texturas. Para a modelagem inicial, utilizamos o software Autodesk Maya.

Para o curta *Fio*, se optou pelo uso do motor gráfico Unreal 4, para obter o render em tempo real. Para ser mais eficiente para o *engine*, o trabalho de modelagem é diferente em alguns aspectos. A modelagem básica, também chamada de *hard surface*, ou de superfície

dura, continua a mesma, mas a malha não é suavizada, aumentando o número de divisões.



Figura 6: banco com *normal map*

Para suavizar a malha, após a modelagem básica, esculpimos o modelo separadamente, como uma outra versão, chamada de *high-poly*, ou de alta poligonagem. É feita uma fusão entre a modelagem básica e a de alta poligonagem, como se o software comparasse as duas geometrias, para criar uma imagem onde os detalhes são impressos em cada polígono, em um mapeamento das normais a cada polígono, no processo que é chamado de *normal map*, mostrado na figura 6.

Alternativamente, para elementos mais orgânicos na cena, como personagens ou árvores, foi usado o software zBrush. Nele, inicialmente é feita a blocagem com formas básicas, assim como na modelagem do tipo *hard surface*, usando pincéis digitais para inserir geometrias, como se fossem cinzéis. Um exemplo de blocagem está na figura 7.



Figura 7: blocagem no zBrush do personagem

Em seguida, é utilizada a ferramenta DynaMesh, para unir todas as partes da blocagem e iniciar um detalhamento mais completo. Com um resultado satisfatório, removemos o DynaMesh, apagando o histórico de alterações, e utilizamos o zRemesher para deixar a malha mais consistente, refinando o detalhamento e preparando para iniciar o processo de modificação da topologia do objeto dentro do Maya.

O processo de modificação da topologia, chamado de retopologia, é iniciado no Maya com a ferramenta Quad-Draw, para criar cada polígono com quatro vértices em cima da malha esculpida, mantendo a forma e deixando a malha menos densa, como a comparação da figura 8.

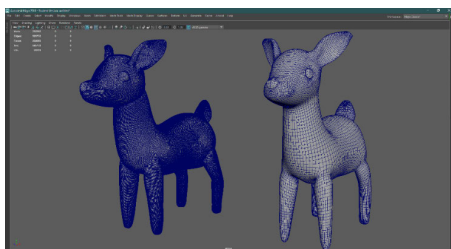


Figura 8: comparação da retopologia do personagem

### Desenvolvimento do visual (*look-dev*)

Na etapa do desenvolvimento do visual, aplica-se os materiais e as texturas nas superfícies dos modelos tridimensionais e se define como a luz interage com estas superfícies. Após a retopologia, se inicia o processo de mapeamento UV. Nesta etapa, criamos uma representação do modelo tridimensional em um plano bidimensional, fazendo cortes na malha de polígono. Estes cortes na malha são colocados no plano 2D, com as ordenadas U e V, daí o nome de mapeamento UV. Na figura 9 pode ser visualizado os pedaços da malha no plano UV.

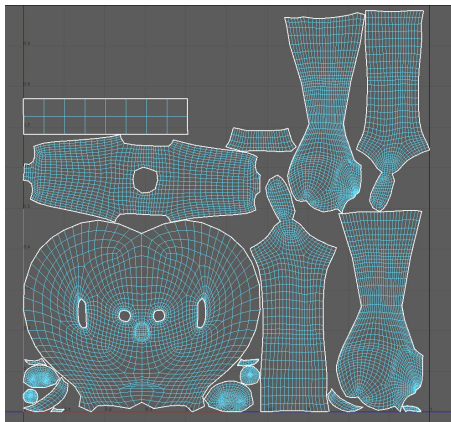


Figura 9: pedaços da malha de polígonos no plano UV

Sobre estes pedaços são aplicadas as texturas previamente construídas. Quando se remonta a malha, as texturas são automaticamente aplicadas na superfície do modelo tridimensional.

Em *Fio*, foram implementadas técnicas de texturização com giz pastel seco, aonde foram criadas as texturas manualmente em papeis próprios para pastel, e posteriormente escaneadas, editadas e organizadas de acordo com o layout do mapeamento UV. Também foi utilizado o software de pintura de textura Substance Painter, para criar mapas de normais aos polígonos das superfícies, determinando como a luz interage com a superfície, para simular alguma forma de relevo, por exemplo. Na figura 10 é visualizado a interface do Substance, com a textura aplicada.

Como toda lã possui fios avulsos saindo dela, foi necessário replicá-los nos animais e nos novelos de lã. Para isso a ferramenta *Xgen* do Maya foi usada, para gerar planos aleatoriamente sobre a superfície. Similarmente, para a folhagem das árvores foi criada uma base com uma malha qualquer, e depois usamos a ferramenta de *Nanomesh*, do *zBrush*, para criar um plano em cada face da base criada. Dessa forma, são criadas texturas com um canal *alpha*, que define a transparência, para que os planos tomem a forma desejada, como na figura 11.

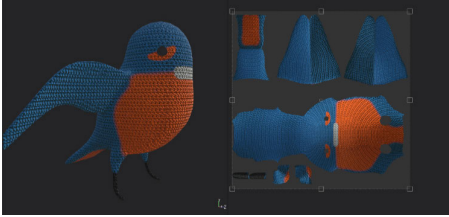


Figura 10: exemplo de texturização no Substance Painter



Figura 11: árvore com planos, no zBrush

Devemos importar todas estas informações para a Unreal Engine, antes de começar a render. A Unreal tem o seu editor de materiais procedurais, que é onde definimos, com cálculos matemáticos, as propriedades físicas das superfícies e volumes encontrados no mundo 3D. A interface intuitiva facilita na definição desses materiais. Os materiais possuem diferentes parâmetros, dependentes de sua configuração predefinida, como materiais translúcidos, com parâmetros de refletividade e de refração, e diversas outras características da física da superfície e de sua interação com a luz.

Nas gramas e árvores, foram usadas máscaras para controlar a composição da textura sobre a superfície. Nestas máscaras, o preto do mapa de textura é definido como transparente e o branco como opaco. Diferente da predefinição de translucência, a máscara não nos permite diferentes níveis de transparência, só invisível ou visível.

Já na pele dos personagens foi utilizada a predefinição de espalhamento na subsuperfície (*subsurface scattering*), que é como se define a penetração da luz em superfícies, para simular comportamentos de interação da luz com a pele ou panos, onde conseguimos ver a luz atravessar a superfície.

## Rigging

Ao mesmo tempo em que é desenvolvido o *look-dev*, começa a ser feito o rigging. O rigging é o procedimento de se criar um esqueleto para o personagem, que vai atuar sobre a sua malha de polígonos, deformando-a para simular o movimento.

Criamos cadeias de *joints*, que são as articulações do personagem, acompanhando a sua anatomia óssea. É definida uma hierarquia entre estas articulações, com a região da bacia sendo o nó mais alto desta hierarquia, para que os ossos (*bones*) sigam um padrão de movimentação entre eles. Na figura 12 se vê um exemplo do esqueleto do personagem principal do *Fio*.

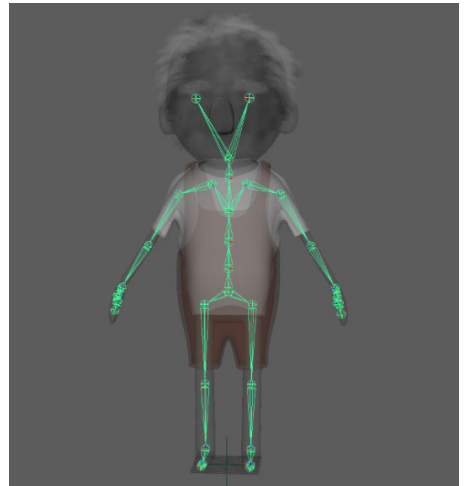


Figura 12: cadeia de articulações no esqueleto, para o rigging

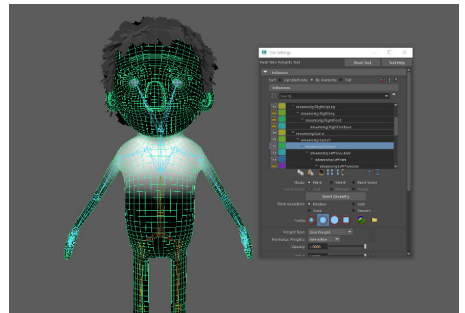


Figura 13: pintura da malha de polígonos no skinning

Com o rigging aplicado, começamos o processo conhecido como *skinning* do personagem. Nesta etapa, é definida a intensidade da atuação do esqueleto para deformar a malha de polígonos. Isto é determinado definindo uma distribuição de peso de atuação sobre a malha, como se fosse uma pintura de peso diretamente na malha, mostrado na figura 13.



Antes ou depois do *skinning*, são criados os controles das articulações, para facilitar o processo de animação, atribuindo uma curva a cada *joint*, com o intuito de que o animador reconheça a função de cada uma, como na figura 14.

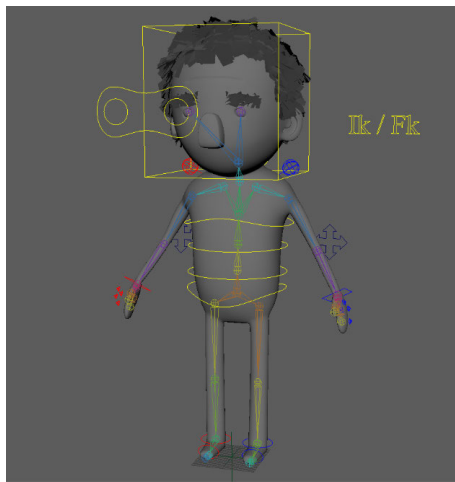


Figura 14: curvas de controle de personagem

Dependendo da situação, é utilizada a cinemática inversa (IK) para facilitar a movimentação dos membros, com a base da hierarquia de joints atuando sobre as articulações mais altas na hierarquia. Por exemplo, para segurar um copo, a mão controla o movimento de todas as articulações do braço.

A outra forma utilizada de cinemática é a direta (FK), onde controlamos a posição e a rotação de cada joint na cadeia, com os elementos seguindo a sequência direta da hierarquia, com os elementos superiores atuando sobre os inferiores na cadeia hierárquica.

De acordo com as necessidades da animação, podemos decidir em usar IK ou FK nas diferentes partes do corpo. É necessário criar um rigging adaptável, que nos permita alternar entre eles. Para isso, criamos três cadeias de joints para cada membro. Uma principal, para o *skinning*, e as outras para o IK e FK, relacionando a cadeia principal com as duas cadeias cinemáticas, para que a principal siga os movimentos de acordo com a preferência por IK ou FK. Para alternar entre IK e FK, criamos um atributo que nos permite escolher entre os dois.

Para finalizar o processo de *rigging*, devemos corrigir os erros que ocorrem naturalmente na atuação do esqueleto sobre a malha, durante os movimentos, como na figura 15. São utilizados os chamados *blend-shapes*, que são variações da malha, com deformações poligonais, que podem ser animáveis, criando uma animação diretamente na malha. Dessa maneira, é usado o editor de poses, para criar correções das poses com *blend-shapes*, que são animados automaticamente de acordo com as movimentações dos joints.



Figura 15: exemplo de correção de pose

## Animação

Logo após o processo de rigging, iniciamos as animações de uma forma simples e não acabada, para termos uma pré-visualização da produção. Uma animação, como todo vídeo, é criada a partir de várias imagens em sequência, em uma velocidade específica, dentro do período de um segundo, que é chamado de quadros por segundo, ou *frames per second* (fps). Para cinema, normalmente é usada a cadência de 24 fps. Para economia de tempo de produção ou por motivos artísticos, 12 quadros por segundo é frequentemente utilizado.

Quanto maior a quantidade de frames por segundo, mais suave a animação, porque são criados uma quantidade maior de *inbetweens*, que são os quadros de preenchimento entre as poses chave da animação. Os quadros intermediários, ou *inbetweens*, podem ser criados automaticamente nos softwares de animação 3D, por meio de técnicas de interpolação. Essas técnicas fazem cálculos de aproximação entre os valores das duas poses chave ao seu redor, que podem ser ajustadas de acordo com as preferências do animador.

Em "Fio" a escolha foi de empregar uma cadência de 12 fps, para se assemelhar com uma animação em stop-motion. Além disso, após finalizadas as poses-chave, transformamos os *inbetweens* criados pelo software em permanentes. É utilizada o tipo de interpolação em etapas, onde não são criados frames inbetween nos quadros decimais e só são lidos os quadros de números inteiros gerados pelo cálculo de *bake*. Assim, o resultado tem uma similaridade maior com a animação em stop-motion — sem ter que animar cada quadro manualmente.

Para a animação dos fios, foram criadas curvas para serem o guia de um cilindro, que representa o fio. Com o deformador *curve warp*, um objeto pode movimentar e se deformar de acordo com a curva desenhada na tela. Para animações em que o fio se deforma, foi criada uma curva curta e se animou seus vértices. O cilindro segue sua animação e nos permite fazer a movimentação do fio pelo espaço.

Nas animações faciais e nas correções finais foram criados *blend-shapes*, que são animados manualmente. Os controladores do rigging disponibilizam uma maior liberdade para deformação

da geometria, para atingir o resultado desejado mais facilmente, como mostrado na figura 16.

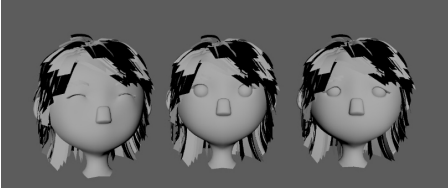


Figura 16: expressões faciais da filha

Após finalizado o processo de animação no Maya, nós a importamos para a Unreal Engine. Primeiramente é exportado o T-Pose de cada personagem que foi animado. Este é o modelo estático do personagem, com os braços abertos e a bacia voltada para o sentido positivo do eixo z, pronto para ser animado.

Depois deste processo, as animações de cada cena são exportadas e conectadas às suas respectivas *Skeletal Meshes*, para que qualquer alteração feita nela seja aplicada nas animações relacionadas. Para personagens, sempre é usada a ferramenta Game Exporter do Maya, que facilita o trabalho com as opções básicas de formatos de arquivo para exportação, como o FBX. A interface do Game Exporter está na figura 17. É importante fazer a seleção correta do que será exportado, como a hierarquia da cadeia de joints, para gerar os frames de cada um individualmente, e as malhas do modelo para receber informação dos Blend-Shapes.

Para animações com deformadores que não necessitam de rigging, como objetos que usam simulação física, ou os fios que seguem curvas, é necessário o uso de um cache de geometria, que guarda informação de transformação de cada vértice e reduz o número de cálculos usados no software. Para isso o Unreal nos permite usar o Alembic, que faz justamente isso, assim nos possibilitando o uso de animações mais complexas sem terem que ser feitas manualmente.

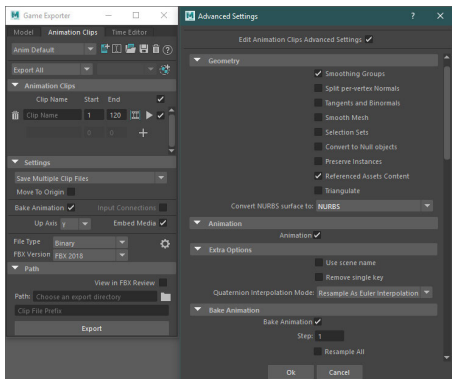


Figura 17: interface do Game Exporter do Maya, com suas configurações

## Iluminação e Render

A etapa de iluminação e do render são feitas diretamente no Unreal, depois que os cenários e personagens são importados. As definições de posicionamento, enquadramento e movimento das câmeras são definidas no Maya e importadas para o motor gráfico também.

O processo de iluminação é essencial para criar a ambientação, além de também ser útil para destacar elementos importantes na narrativa. Uma das maiores dificuldades encontradas foi de diferenciar as quatro estações do ano ao longo das quais a estória acontece. Além dos aspectos técnicos encontrados no processo de iluminação, existe também uma ênfase na representação artística e subjetiva dos sentimentos representados pelos personagens, como foi concebido durante o curta, como na figura 18.



Figura 18: cena que acontece no inverno, com as luzes e o fog

Dentro do Unreal foram utilizados diversos tipos de luz, comumente encontrados nos softwares de animação e render, como exibido na figura 19. A luz direcional foi utilizada como luz principal, gerando uma iluminação em todo o cenário a partir de uma direção escolhida pelo usuário. Ela é usada para representar a luz do sol, por exemplo. A *spot light*, que é uma iluminação gerada em um cone, é usada para dar ênfase em algo ou provocar raios de luz mais controlados. Existe também o *area light*, que é uma luz criada a partir de uma área retangular, para fins gerais e específicos, e o *point light*, que é uma luz gerada a partir de um ponto em uma área limitada ao seu redor. Para o curta Fio, foi utilizado o efeito de névoa, que é o *fog*, para criar raios de luz e seus espalhamentos.



Figura 19: cena em que foi utilizada a maior quantidade de luzes diferentes

A equipe deste projeto tem experiência com diferentes renderizadores, como o Arnold, o Mental Ray e o RenderMan. Os resultados obtidos, com a configuração adequada, são realistas e de alta

qualidade, mas demandam um tempo de render que levaria mais de um dia para renderizar todo o curta metragem, cujo tempo de duração ficou em torno de cinco minutos. No entanto para o tempo disponível para este trabalho, era necessário testar as soluções experimentais em um intervalo que permitisse finalizar todo o trabalho em quatro meses. Desta forma, foi escolhido o Unreal Engine, pela qualidade do resultado final e pelo tempo gasto para renderizar cada quadro de imagem, quase em tempo real.

As cenas finalizadas no Maya, com texturas, materiais, animações e câmeras foram exportadas para o Unreal, no formato .fbx. O Unreal tem suporte para este formato, que tem a informação completa de cada cena.

No motor gráfico, foram verificadas cada uma das texturas e, caso fosse necessário, alterados seus parâmetros para terem o mesmo visual que no Maya. Em seguida, com as câmeras e a iluminação definidas, são feitos os ajustes dos parâmetros para o render, levando em consideração cada textura, os sombreadores, os materiais, as reflexões, a iluminação estática e a dinâmica, as transparências, para gerar a saída da imagem. A grande vantagem em todo este processo, é que o resultado final é rapidamente visualizável e as modificações para melhorar cada cena podem ser testadas imediatamente.

O formato dos quadros de saída foi definido como o .exr. Este formato permite um intervalo dinâmico alto para as cores e os brilhos, além de conter informações para profundidade de campo e outros parâmetros importantes para a pós-produção.

## Pós-produção

Com os frames da animação com a saída .exr, foi realizada a edição, a correção e gradação de cor e a sonorização da animação, na etapa da pós-produção. Neste momento, o fluxo de trabalho voltou a ser o mesmo utilizado em toda a indústria criativa.

A gradação de cor teve uma atenção especial, para reforçar a ambiência e as sensações que a direção de arte buscou abordar. Para atingir o resultado, foram alterados o contraste e a saturação, a temperatura de cor, e foram feitos ajustes de exposição e de controle de nitidez e desfoque. Grande parte deste trabalho foi realizado diretamente na Unreal previamente, e corrigidos no After Effects, quando havia necessidade.



Figura 20: antes da correção e gradação de cor, à esquerda, e depois, à direita

## Conclusão

A incorporação do motor gráfico para jogos digitais, o Unreal Engine, para realizar diversas etapas da produção do curta metragem de animação, trouxe uma série de melhorias na realização do trabalho. As soluções do Unreal para renderizar de forma fotorealista as imagens, e em tempo real, permite que uma etapa crítica da produção seja feita sem tantos requisitos de hardware, como era há alguns anos atrás.

No fluxo de trabalho realizado com os renderizadores existentes até há cinco anos atrás, seria necessário processamento central e gráfico e quantidade de memória RAM elevados, gerando custos adicionais, além de um tempo para a realização do render de muitas horas. Defeitos aparecem no render final e eles têm que ser corrigidos e o render refeito. Isto demanda dias de trabalho, ou até meses, dependendo da complexidade das cenas e da duração da animação.

Com a evolução das soluções para fazer as etapas do render, como obtidas pelo motor Unreal, estes problemas foram superados e imagens com realidade fotorealistas são geradas em um intervalo de tempo muito curto, inimagináveis há cinco anos atrás.

Superadas as dificuldades para importar os arquivos do Maya para o Unreal, e tendo um controle e conhecimento maior dos parâmetros envolvidos no processo de renderização, as vantagens apresentadas pelo render em tempo real fazem com que ele seja uma tendência que certamente se tornará o fluxo de trabalho mais comum a ser adotado nos próximos anos.

O vídeo do primeiro minuto do curta metragem Fio está em <https://vimeo.com/503536788>.

## Agradecimento

Os autores gostariam de agradecer o apoio do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), do Brasil, para a realização deste trabalho.

## Referências

- A Boy and His Kite*, 2015. De Gavin Moran. Estados Unidos: Epic Games.
- Akenine-Möller, Tomas, Haines, Eric, Hoffman, Naty. 2018. Real Time Rendering. A K Peters/CRC Press. 4<sup>th</sup> Edition.
- Fio*, 2020, De Igor Arantes, Samuel Leidner e Bernardo Bracarense. Brasil.
- Hearthstone Animated Short: Win or Lose*, 2019. De Brian Horn. Estados Unidos: Blizzard Entertainment.
- Lost and Found*, 2018. De Bradley Slabe e Andrew Goldsmith. Austrália.
- Monsters S.A. (Monsters, Inc.)*, 2001. De Pete Docter. Estados Unidos: Pixar.
- O Menino e o Mundo*, 2013. De Alê Abreu. Brasil: Filme de Papel.
- Revolving Rhymes*, 2016. De Jakob Schuh e Jan Lachauer. Alemanha e Reino Unido.



*Toy Story*, 1995. De John Lasseter. Estados Unidos: Pixar.

*Up: Altas Aventuras (Up)*, 2009. De Pete Docter. Estados Unidos: Pixar.

UNREAL Engine. 2015. SIGGRAPH Recognizes 'A Boy and His Kite' with Best Real-Time Graphics and Interactivity Award. <https://www.unrealengine.com/en-US/blog/siggraph-unreal-kite-best-real-time-graphics-interactivity-award>, acessado em 25 de abril de 2020.

UNREAL Engine 2019, Unreal Studio 4.22 advances real-time photoreal rendering. <https://www.unrealengine.com/en-US/blog/unreal-studio-4-22-advances-real-time-photoreal-rendering>, acessado em 23 de abril de 2020.

VASEK, Milan. 2020. *Autumn Lake*, disponível em Artstation, <https://www.artstation.com/artwork/AXbxV>, acessado em 23 de abril de 2020.

*Viva: a Vida É uma Festa (Coco)*, 2017. De Lee Unkrich e Adrian Molina. Estados Unidos: Pixar.