

Tema:

Shaders de pós-processamento visual open-source para Unreal Engine 5

Contexto e Motivação

No contexto de jogos digitais, a produção de conteúdo que acaba sendo descartado ao longo do projeto é um dos fatores que onera o tempo de um projeto. Dentre os processos que sofrem com esse aspecto, está a arte técnica, área relacionada à produção de códigos para a renderização de elementos gráficos do jogo (conhecidos como Shaders). Essa área requer profissionais especializados versados tanto no campo da arte quanto no campo da programação e, portanto, trabalho desses profissionais deve ser cautelosamente levado em consideração durante a produção. Evitar que tempo de produção seja gasto em elementos que posteriormente serão descartados pode se mostrar um fator crucial na finalização de projetos longos. Isso se mostra verdade sobretudo no contexto da indústria de jogos brasileira, que ainda é composta em sua maioria por pequenas empresas independentes.

Portanto, este trabalho se propõe a trazer uma solução para parte desse problema, focando na área de Shaders de pós-processamento (efeitos visuais aplicados a uma cena já renderizada). A solução envolve disponibilizar um Asset Pack contendo Shaders de pós-processamento open-source feitos para projetos de jogos e experiências digitais. Com isso, será possível testar diferentes efeitos visuais prontos dentro de diferentes projetos sem que seja necessário que o artista técnico gaste tempo produzindo esses Shaders, especialmente em fazer iniciais do projeto. Assim, a direção artística do jogo pode ser encaminhada e averiguada mais cedo e com menos trabalho sendo desperdiçado.

Objetivos

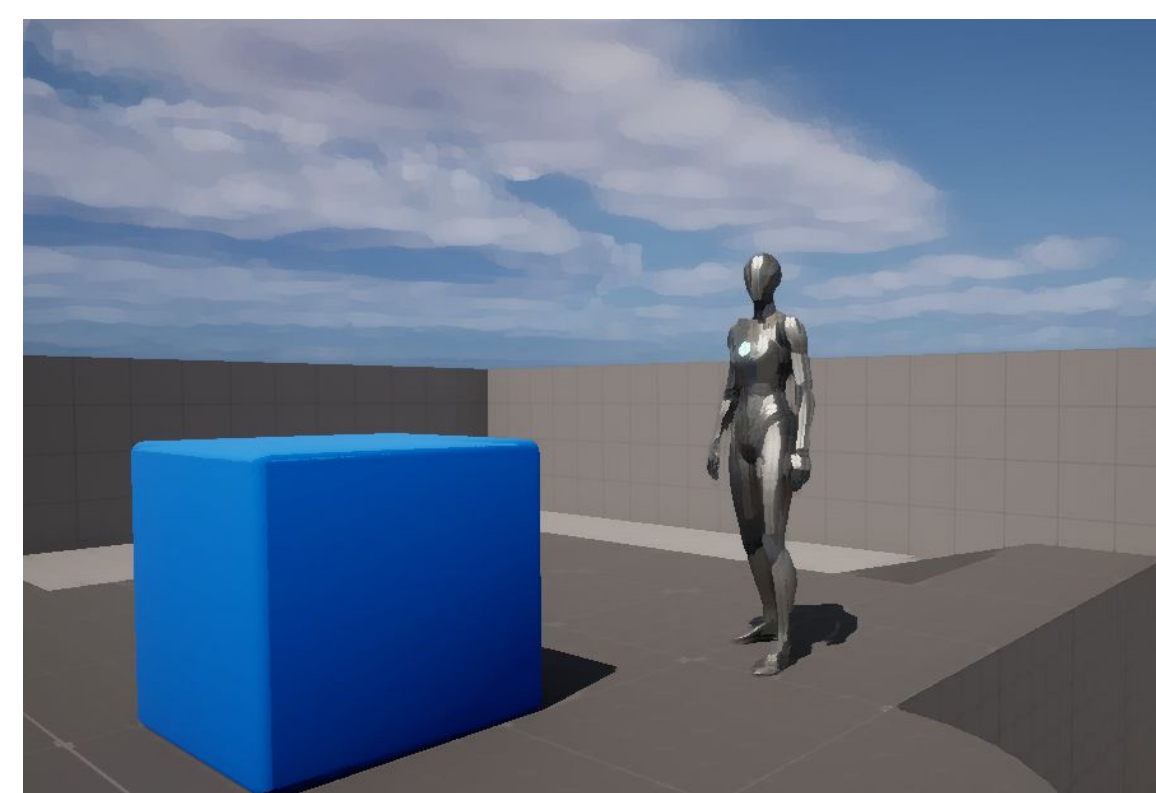
- Produzir um Asset Pack para a Unreal Engine 5 com pelo menos 5 diferentes efeitos visuais de pós-processamento e disponibilizá-lo gratuitamente
- Assegurar que os efeitos produzidos não afetem significativamente a performance de projetos

Metodologia

Na produção dos efeitos, foram utilizadas as tecnologias disponíveis no motor de jogos "Unreal Engine 5", bem como a linguagem de programação de Shaders desenvolvida pela Microsoft "HLSL" (High Level Shader Language). Para tanto, foi necessário estudo e pesquisa a respeito do funcionamento da renderização em jogos digitais e da arquitetura básica de GPUs modernas. Para averiguar a performance num ambiente similar a de um projeto real, cada um dos efeitos produzidos foi também testado utilizando projetos completos disponibilizados

gratuitamente pelos desenvolvedores da Epic, os chamados "Sample Games". Foram medidos em cada caso a quantidade de frames renderizados por segundo e o impacto causado na thread da GPU.

Efeitos Implementados



"Filtro Kuwahara", que adiciona à cena um efeito similar ao de uma pintura à óleo.

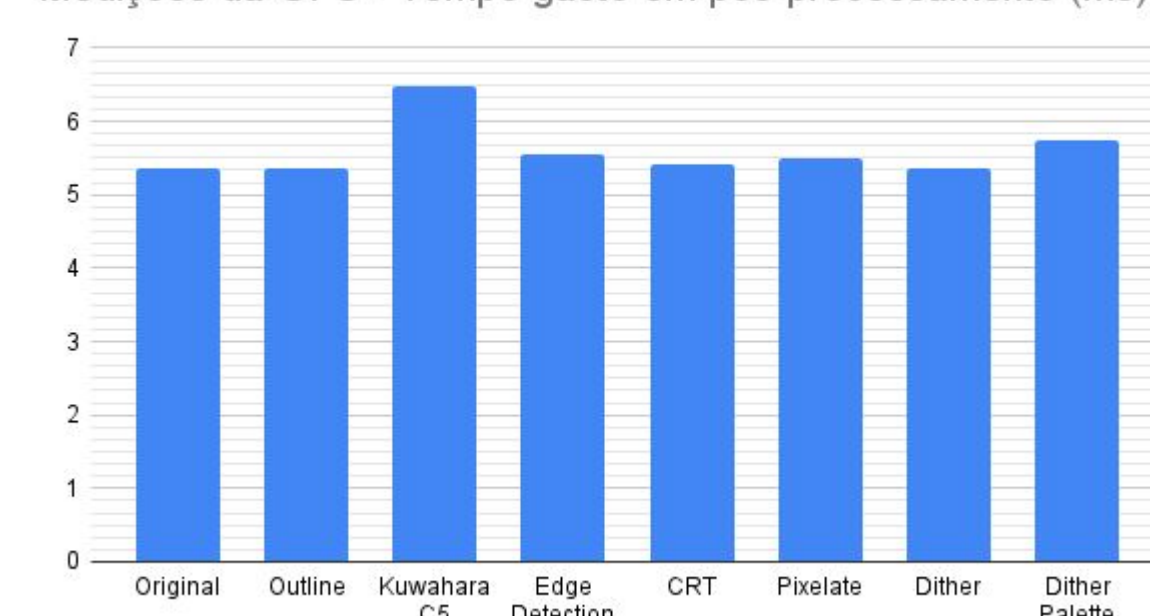


Efeito do "CRT Filter", que deixa a cena com um visual semelhante a imagens em televisões de tubo (CRT).

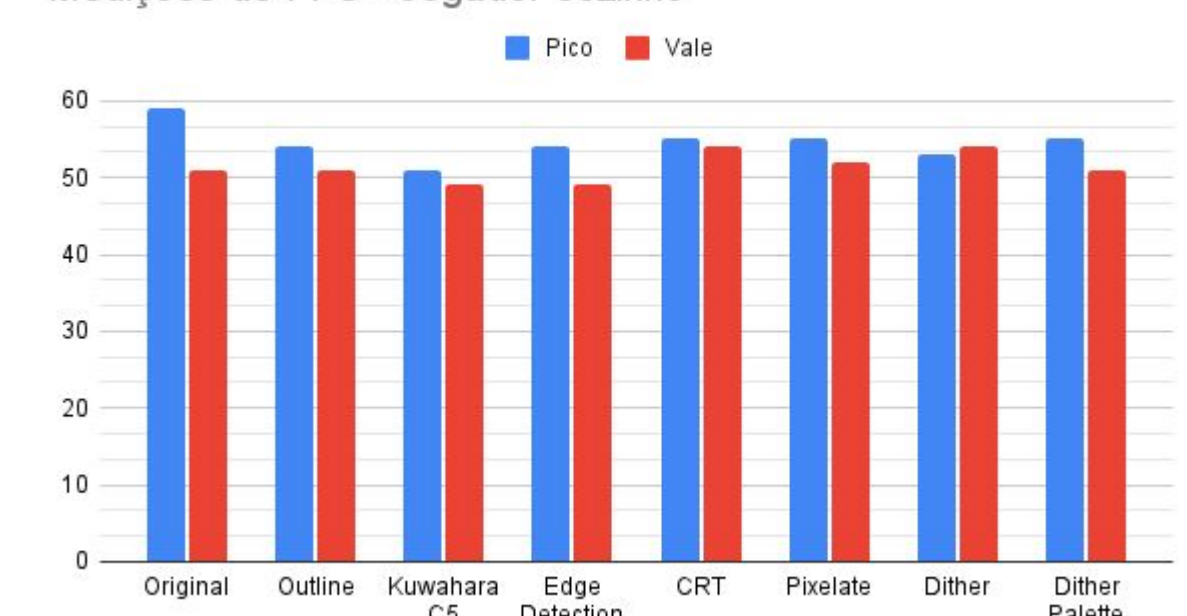
Ao todo, foram implementados 6 diferentes efeitos visuais: **"Dithering"**, **"Kuwahara Filter"**, **"Outline"**, **"Edge Detection"**, **"Pixelate"** e **"CRT Filter"**. Cada um deles foi escolhido para trazer um aspecto visual diferente ao projeto.

Testes e Resultados

Medições da GPU - Tempo gasto em pós-processamento (ms)



Medições de FPS - Jogador sozinho



Medições de FPS e tempo de pós-processamento na thread da GPU, medidos utilizando o Sample Game "Lyra"

Apesar de terem havido limitações por conta da capacidade das máquinas utilizadas nos testes, foi possível medir a performance dos Shaders em comparação às medições feitas sem os efeitos produzidos. Os resultados mostraram-se satisfatórios no que diz respeito a causar impactos mínimos à performance do jogo.