



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CENTRO TECNOLÓGICO  
CURSO DE GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

Emily Sayuri Kiba

**Manual Prático de Uso de Inteligência Artificial Gerativa na Criação de  
Modelos 3D para Jogos Digitais**

Florianópolis  
2025

Emily Sayuri Kiba

**Manual Prático de Uso de Inteligência Artificial Gerativa na Criação de  
Modelos 3D para Jogos Digitais**

Trabalho de Conclusão de Curso do Curso de Graduação em Ciências da Computação do Centro Tecnológico da Universidade Federal de Santa Catarina para a obtenção do título de bacharel em Ciências da Computação.  
Orientador: Prof. Aldo von Wangenheim, Dr. rer. nat.

Florianópolis  
2025

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.  
Dados inseridos pelo próprio autor.

Kiba, Emily Sayuri

Manual prático de uso de inteligência artificial  
gerativa na criação de modelos 3D para jogos digitais /  
Emily Sayuri Kiba ; orientador, Aldo von Wangenheim, 2025.  
79 p.

Trabalho de Conclusão de Curso (graduação) -  
Universidade Federal de Santa Catarina, Centro Tecnológico,  
Graduação em Ciências da Computação, Florianópolis, 2025.

Inclui referências.

1. Ciências da Computação. 2. Inteligência artificial  
gerativa. 3. Modelagem 3D. 4. Jogos digitais. I.  
Wangenheim, Aldo von. II. Universidade Federal de Santa  
Catarina. Graduação em Ciências da Computação. III. Título.

Emily Sayuri Kiba

**Manual Prático de Uso de Inteligência Artificial Gerativa na Criação de Modelos 3D para Jogos Digitais**

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de “bacharel em Ciências da Computação” e aprovado em sua forma final pelo Curso de Graduação em Ciências da Computação.

Florianópolis, 10 de Dezembro de 2025.

---

Profa. Lúcia Helena Martins Pacheco, Dra.  
Coordenador do Curso

**Banca Examinadora:**

---

Prof. Aldo von Wangenheim, Dr. rer. nat.  
Orientador

---

Prof. Flávio Andaló, Dr.  
Avaliador  
Universidade Federal de Santa Catarina

---

Eduardo Beckhauser.  
Avaliador  
Universidade Federal de Santa Catarina

## AGRADECIMENTOS

Agradeço à minha mãe, Miyuki, e ao meu pai, Takuma, por sempre me apoiarem e me proporcionarem oportunidades para chegar até aqui. Obrigada por acreditarem em mim e me incentivarem, tudo isso sem pressão, com paciência e muito carinho.

Aos professores do curso de Ciências da Computação, que me proporcionaram tanto conhecimento. Principalmente o professor Aldo, pela orientação, oportunidades e contribuições valiosas ao longo do meu TCC. Agradeço também ao professor Flávio, que me forneceu a modelagem da Ilha de Anhatomirim, essencial para o desenvolvimento do meu TCC. A base fornecida serviu como ponto de partida para a criação do jogo, permitindo realizar alterações e adaptações que enriqueceram o projeto final.

Ao meu namorado, Rafael, por estar no meu lado me apoiando e aliviando minha ansiedade e o estresse. Pelos momentos de diversão e paz que me permitiram respirar e recarregar as energias, pelas nossas "sessões de MasterChef" cozinhando juntos e rindo, e pelas discussões sobre filmes.

Aos meus três gatos – Mei, Cookie e Low – por estarem sempre dormindo de forma tão confortável que me dava vontade de abandonar tudo e deitar junto. Ou por me lembrarem que era hora de levantar para servir comida.

À minha prima, que durante a sua graduação na UFSC dividiu comigo tantos momentos do dia a dia, agradeço pela companhia e pelas conversas em casa. Aos meus avôs, que apareciam para visitar e alegrar o ambiente. Em especial à minha avó, que sempre ajudou nas tarefas da casa, apesar de gostar de me contar muitas fofocas repetidas em todos os momentos, tornou a casa cheio de risadas.

Agradeço também aos meus amigos da Ciências da Computação – Eric, Bruno, Gabriela, Gabriel, Julien, Samantha e Marco – pelos inúmeros momentos de diversão, risadas e experiências inesquecíveis. Pelas conversas sobre matérias e professores, e pelos roles como karaokês, trilhas e aniversários, que tornaram o curso muito mais leve e divertido.

Especialmente o Marco, que conheci no início do curso, durante a pandemia, e que sempre foi meu parceiro de trabalho confiável. Enfrentamos várias madrugadas e trabalhos complicados, mas sobrevivemos junto.

Ao Gabriel, mesmo tendo nos conhecido nos últimos dois anos, nossa amizade parece de longa data. As conversas que nunca têm fim – sobre estudos, nossos hiperfocos, discussões animadas, filmes e séries – tornaram a reta final do curso muito mais divertida e significativa.

Por fim, agradeço a todos que, de alguma forma, fizeram parte desta caminhada. Cada gesto de apoio, palavra de incentivo, conversa e risada contribuíram para que este trabalho se tornasse realidade.

## RESUMO

Este trabalho explora a aplicação de Inteligência Artificial (IA) gerativa, especificamente o Stable Diffusion, na área de modelagem 3D e sua integração no contexto de jogos. O estudo tem como foco a criação de manual prático detalhado, demonstrando como a IA gerativa pode ser utilizada tanto para gerar ideias e conceitos visuais para modelos 3D quanto para a criação direta de malhas e texturas tridimensionais. Dessa forma, o trabalho evidencia o potencial da IA em agilizar o processo de modelagem 3D, simplificando-o ao integrar a IA como uma ferramenta de suporte à criação, tornando-o mais acessível especialmente para usuários sem experiência prévia em modelagem, e contribuindo para a criatividade e eficiência na produção de ativos digitais para jogos.

**Palavras-chave:** Inteligência Artificial Gerativa. Stable Diffusion. Modelagem 3D. Game Engine.

## ABSTRACT

This work explores the application of Generative Artificial Intelligence (AI), specifically Stable Diffusion, in 3D modeling and its integration into the context of games. The study presents a detailed practical manual demonstrating how generative AI can be used both to generate visual ideas and concepts and to directly create 3D meshes and textures. In doing so, it highlights the potential of AI to streamline and simplify the 3D modeling process, integrating it as a supportive tool for creation, making it accessible to users without prior experience and enhancing creativity and efficiency in the production of digital assets for games.

**Keywords:** Generative Artificial Intelligence. Stable Diffusion. 3D Modeling. Game Engine.

## LISTA DE FIGURAS

Figura 1 – Linha do tempo da história de IA. . . . .	21
Figura 2 – Arquitetura de GANs. . . . .	22
Figura 3 – Passos para adição e remoção de ruído. . . . .	23
Figura 4 – Exemplo de text-to-image. . . . .	25
Figura 5 – Exemplo de image-to-image. . . . .	26
Figura 6 – Exemplo de inpainting. . . . .	26
Figura 7 – Exemplos de ControlNet. . . . .	27
Figura 8 – Exemplos de LoRA. . . . .	28
Figura 9 – Interface de Automatic1111. . . . .	30
Figura 10 – Interface de ComfyUI. . . . .	31
Figura 11 – Exemplo de geração de textura - ComfyUI. . . . .	33
Figura 12 – Exemplo de geração de imagem - ComfyUI. . . . .	34
Figura 13 – Exemplo prático de inpainting do manual. . . . .	35
Figura 14 – Exemplo prático de ControlNet Scribble do manual. . . . .	35
Figura 15 – Exemplos de resultados de LoRA do manual. . . . .	36
Figura 16 – Pipeline e arquitetura do Hunyuan 3D 2.0. . . . .	38
Figura 17 – Exemplo de Hunyuan 3D 2.0 no ComfyUI . . . . .	39
Figura 18 – Visão geral de geração de materiais no Hunyuan 3D 2.5 . . . . .	40
Figura 19 – Exemplo de uso de Hunyuan 3D 2.5 . . . . .	40
Figura 20 – Exemplo de uso de Meshy AI . . . . .	41
Figura 21 – Comparação do resultado entre 3 modelos . . . . .	42
Figura 22 – Exemplo de malha colada e malha corrigida no Blender . . . . .	43
Figura 23 – Exemplo de textura incompleta e corrigida no Blender . . . . .	43
Figura 24 – Exemplo de ajuste de Shader no Blender . . . . .	44
Figura 25 – Blender - pintura de pesos . . . . .	45
Figura 26 – Hunyuan 3D 2.5 - gemas . . . . .	47
Figura 27 – Hunyuan 3D 2.5 - plataformas . . . . .	47
Figura 28 – Hunyuan 3D 2.5 - grade de madeira . . . . .	48
Figura 29 – Unreal Engine 5 - Blueprint de gemas . . . . .	48
Figura 30 – Unreal Engine 5 - Blueprint de plataforma . . . . .	50
Figura 31 – Unreal Engine 5 - Blueprint de grade de madeira (função de verificação) . . . . .	50
Figura 32 – Unreal Engine 5 - Comparação entre tiling e bombing . . . . .	52
Figura 33 – Unreal Engine 5 - Mesclagem de Textura . . . . .	52
Figura 34 – Personagem inspirada em Maricota . . . . .	53
Figura 35 – Ambiente do jogo - Fortaleza de Anhatomirim portada . . . . .	53
Figura 36 – Ambiente do jogo - Fortaleza de Anhatomirim casa do comandante . . . . .	54
Figura 37 – Ambiente do jogo - Fortaleza de Anhatomirim igreja . . . . .	54

Figura 38 – Ambiente do jogo - Fortaleza de Anhatomirim ponta do quartel . . . .	55
Figura 39 – Ambiente do jogo - Fortaleza de Anhatomirim na ponta . . . . .	55

## LISTA DE TABELAS

Tabela 1 – Resumo de aplicações de ferramentas . . . . .	18
Tabela 2 – Comparação entre principais interfaces WebUI para Stable Diffusion .	29
Tabela 3 – Comparativo entre Automatic1111 e ComfyUI. . . . .	31

## LISTA DE ABREVIATURAS E SIGLAS

GANs	<i>Generative Adversarial Network</i>
IA	Inteligência Artificial
LoRA	<i>Low-Rank Adaptation</i>
PBR	<i>Physically Based Rendering</i>
VAE	<i>Variational Autoencoder</i>
VRAM	<i>Video Random Access Memory</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>13</b>
1.1	OBJETIVOS	13
1.1.1	<b>Objetivo Geral</b>	<b>14</b>
1.1.2	<b>Objetivos Específicos</b>	<b>14</b>
1.2	METODOLOGIA	14
<b>2</b>	<b>ESTADO DA ARTE</b>	<b>17</b>
<b>3</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>19</b>
3.1	DEFINIÇÃO E CONCEITOS DE IA GERATIVA	19
3.2	CONTEXTO HISTÓRICO	19
3.3	PRINCIPAIS ABORDAGENS DE IA GERATIVA	22
3.3.1	<b>Redes Adversariais Generativas (GANs)</b>	<b>22</b>
3.3.2	<b>Modelos de Difusão</b>	<b>23</b>
3.4	TECNOLOGIAS DA IA GERATIVA	24
3.4.1	<b>Stable Diffusion</b>	<b>24</b>
3.4.2	<b>Text-to-Image e Image-to-Image</b>	<b>24</b>
3.4.3	<b>Inpainting</b>	<b>26</b>
3.4.4	<b>ControlNet</b>	<b>27</b>
3.4.5	<b>LoRA</b>	<b>28</b>
3.5	FERRAMENTAS UTILIZADAS	29
3.5.1	<b>Automatic1111</b>	<b>29</b>
3.5.2	<b>ComfyUI</b>	<b>30</b>
<b>4</b>	<b>DESENVOLVIMENTO DO MANUAL</b>	<b>32</b>
4.1	GERAÇÃO DE IMAGEM	32
4.2	GERAÇÃO E AJUSTE DE MODELOS 3D	37
4.2.1	<b>Hunyuan 3D 2.0</b>	<b>37</b>
4.2.2	<b>Hunyuan 3D 2.5</b>	<b>39</b>
4.2.3	<b>Meshy AI</b>	<b>41</b>
4.2.4	<b>Correção e Ajuste de Modelos 3D</b>	<b>42</b>
4.3	APLICAÇÃO DE ESQUELETOS E ANIMAÇÃO	44
4.4	CRIAÇÃO DE JOGO	46
<b>5</b>	<b>RESULTADOS E LIMITAÇÕES DO USO DE IA GERATIVA</b>	<b>56</b>
5.1	PONTOS FORTES OBSERVADOS	56
5.2	LIMITAÇÕES	57
<b>6</b>	<b>QUESTÕES ÉTICAS E CUIDADOS NO USO DE IA GERATIVA</b>	<b>58</b>
<b>7</b>	<b>CONCLUSÃO</b>	<b>60</b>
	<b>REFERÊNCIAS</b>	<b>61</b>
	<b>APÊNDICE A – RELATÓRIO TÉCNICO</b>	<b>63</b>

<b>APÊNDICE B – MANUAL PRÁTICO . . . . .</b>	<b>65</b>
<b>APÊNDICE C – PROTÓTIPO DE JOGO EM UNREAL ENGINE</b>	<b>67</b>
<b>APÊNDICE D – ARTIGO . . . . .</b>	<b>68</b>

# 1 INTRODUÇÃO

Atualmente, a modelagem tridimensional (3D) desempenha um papel fundamental nas indústrias de entretenimento, como animação, cinema, jogos e realidade virtual, contribuindo para a criação de experiências imersivas ao público. Com o aumento da demanda por conteúdo 3D e a constante evolução do mercado, artistas e designers enfrentam desafios significativos tanto na geração de ideias quanto na criação de texturas realistas que aprimoram a qualidade visual dos objetos. Além desses profissionais, há também um número crescente de desenvolvedores e entusiastas que desejam criar seus próprios jogos, mas que não possuem conhecimento aprofundado em arte digital ou computação gráfica.

Ao mesmo tempo, emergiu uma classe recentemente popular de modelos variáveis latentes de inteligência artificial chamados de modelo de difusão, os quais podem ser usados para várias tarefas, incluindo *denoising* de imagem (reduzir os ruídos), *inpainting* (preenchimento), *upscaling* (aumentar resolução) e geração de imagem, a partir de descrição de texto simples e/ou imagem como a entrada. Essas técnicas ajudaram a popularizar a Inteligência Artificial (IA) gerativa (Lawlor; Chang, 2023), tornando essas ferramentas poderosas para design conceitual em qualquer disciplina que exija criatividade em tarefas de design visual. Isso também se aplica aos estágios iniciais do projeto arquitetônico, envolvendo ideação, esboço e modelagem (Ploennigs; Berger, 2023). As capacidades gerativas dessas ferramentas provavelmente alteraram fundamentalmente os processos criativos pelos quais os criadores formulam ideias e as transformam em produtos (Epstein; Hertzmann, 2023). Diante desses avanços e vantagens notáveis, a IA gerativa tem chamado a atenção da indústria e da comunidade de pesquisa. Seu impacto não se limita apenas aos campos mencionados, mas se estende a setores diversos, como publicidade, entretenimento e educação. Portanto, a adaptação dessas ferramentas inovadoras nas práticas industriais está se tornando uma estratégia importante e promissora.

Neste contexto, este trabalho propõe explorar o potencial da IA gerativa, especialmente o Stable Diffusion, baseado em um tipo particular de modelo de difusão chamado *Latent Diffusion Model*, como uma ferramenta promissora para enfrentar os desafios mencionados e aumentar a eficiência na criação de conteúdo. Pretende-se aplicá-la no desenvolvimento de jogos, explorando sua utilização na criação de ambientes e ativos tridimensionais. Para alcançar resultados satisfatórios, é necessário compreender seu funcionamento e fornecer entradas e configurações corretas, garantindo instruções claras e objetivas para um desempenho otimizado.

## 1.1 OBJETIVOS

Este trabalho tem como objetivo investigar o uso de ferramentas de IA gerativa para apoiar a criação de imagens e modelos 3D. A proposta é demonstrar como essas ferramentas podem auxiliar desenvolvedores na etapa de concepção visual de personagens,

objetos e cenários, além de explorar sua aplicação na geração de modelos 3D a partir de imagens, contribuindo para o processo de desenvolvimento de jogos digitais. Para isso, será elaborado um manual prático, com exemplos, voltado a iniciantes na área.

### 1.1.1 Objetivo Geral

O objetivo geral deste trabalho é investigar e demonstrar como a aplicação de IA gerativa, com foco em Stable Diffusion e suas interfaces Automatic1111 e ComfyUI, pode aprimorar o processo de criação de objeto 3D e suas texturas realistas, fornecendo orientações práticas do uso dela como ferramenta eficaz na modelagem 3D, com sucessivas utilizações na criação de jogo.

### 1.1.2 Objetivos Específicos

- Explorar os conceitos de IA gerativa voltados à geração de imagens e modelos 3D.
- Investigar o funcionamento e os recursos das interfaces Automatic1111 e ComfyUI na geração de imagens.
- Aplicar técnicas de conversão de imagem para modelo 3D com ferramentas compatíveis.
- Desenvolver um manual prático para orientação de usuários iniciantes.
- Avaliar o impacto da IA gerativa no processo criativo, eficiência e produtividade.
- Criar um pequeno protótipo de jogo utilizando elementos criados com auxílio da IA gerativa.

## 1.2 METODOLOGIA

Para alcançar os objetivos específicos apresentados, o estudo será conduzido em várias etapas detalhadas a seguir:

### **Etapa 1 – Estado da Arte**

A primeira etapa consistiu na realização de um estudo do estado da arte, visando analisar estudos, artigos e recursos relacionados à aplicação de IA gerativa na modelagem 3D e em contextos criativos, fornecendo uma base sólida para a compreensão do panorama atual do campo e orientando as etapas subsequentes da pesquisa. O levantamento foi realizado por meio de uma revisão sistemática de literatura, cujos procedimentos metodológicos e resultados detalhados estão documentados em relatório técnico específico.

**Atividade 1.1:** Levantamento bibliográfico e técnico sobre o uso de IA gerativa em contextos criativos, com foco em aplicações práticas e pesquisas recentes.

**Atividade 1.2:** Seleção de artigos e fontes relevantes com base em critérios previamente definidos.

**Atividade 1.3:** Análise e síntese das informações dos artigos selecionados, destacando tendências e ferramentas mais utilizadas na literatura.

## **Etapa 2 – Fundamentação Teórica**

Esta etapa consistiu na realização de uma pesquisa sobre o contexto histórico da IA gerativa e nos conceitos fundamentais das principais abordagens nessa área, fornecendo uma base teórica que permite a compreensão aprofundada dos tópicos abordados nas etapas subsequentes.

**Atividade 2.1:** Levantamento do contexto histórico da IA gerativa, incluindo marcos importantes, evolução tecnológica e contribuições significativas na área.

**Atividade 2.2:** Pesquisa e estudo das principais abordagens em IA gerativa, com destaque para conceitos, técnicas e ferramentas relevantes.

## **Etapa 3 – Explicações técnicas**

Serão apresentadas e detalhadas as principais técnicas utilizadas no processo de geração de imagens e modelos 3D com IA.

**Atividade 3.1:** Explicação sobre técnicas utilizadas na geração de imagens.

**Atividade 3.2:** Explicação sobre técnicas utilizadas na geração de modelos 3D.

## **Etapa 4 – Desenvolvimento de manual prático**

Será desenvolvido um manual prático detalhado, contendo instruções passo a passo sobre o uso do Stable Diffusion e de suas interfaces Automatic1111 e ComfyUI, abordando tanto a geração de ideias e texturas para objetos 3D quanto a criação direta de modelos 3D. O manual demonstrará a aplicação desses objetos em um jogo digital, evidenciando a utilidade prática das ferramentas.

**Atividade 4.1:** Apresentar o uso de Automatic1111 e ComfyUI para geração de imagens.

**Atividade 4.2:** Demonstrar técnicas que permitem maior controle e estilização, como ControlNet e LoRA.

**Atividade 4.3:** Apresentar o uso do ComfyUI para geração de modelos 3D.

**Atividade 4.4:** Demonstrar a correção e aplicação de esqueleto nos modelos 3D.

**Atividade 4.5:** Apresentar passo a passo a criação de um jogo digital utilizando os modelos 3D.

**Etapa 5 – Análise de resultados e considerações éticas**

Após o desenvolvimento do manual e a aplicação prática no jogo, será realizada uma análise dos resultados obtidos, avaliando pontos fortes, limitações e impactos do uso de IA gerativa. Além disso, serão discutidas recomendações para o uso ético, legal e responsável dessas ferramentas, abordando direitos autorais, autoria, originalidade e boas práticas no desenvolvimento de ativos digitais.

**Atividade 5.1:** Apresentar os pontos fortes e limitações das ferramentas de IA gerativa.

**Atividade 5.2:** Pesquisar e apresentar recomendações para o uso ético, legal e responsável da IA gerativa.

## 2 ESTADO DA ARTE

Nesta seção, são destacados os artigos que empregaram IA gerativa de forma eficiente, contribuindo para a modelagem 3D, bem como algumas ferramentas que demonstraram resultados relevantes. Esses artigos foram previamente analisados e sintetizados no relatório técnico (Kiba; Wangenheim, 2024), apresentado no Apêndice A, que serve como base para esta revisão. O relatório proporciona uma visão mais abrangente das aplicações e metodologias recentemente utilizadas na área.

Os artigos selecionados apresentam métodos eficazes para a geração de modelos 3D a partir de entradas de texto ou imagens, como os modelos MVDreams e One-2-3-45++. Ambos os modelos de difusão utilizam visualização múltipla para gerar modelos 3D de forma mais coerente. O MVDreams gera modelos 3D a partir de entradas de texto, enquanto o One-2-3-45++ transforma imagens de entrada em modelos 3D. Além destes, foram selecionados artigos que contribuem para a texturização de modelos 3D, como GenesisTex e Material Palette. O GenesisTex sintetiza texturas para geometria 3D a partir de entrada de texto, enquanto o Material Palette extrai mapas *Physically Based Rendering* (PBR) a partir de imagens de entrada.

Também foram analisadas publicações que apresentam fluxos de trabalho demonstrando a geração de imagens específicas com Stable Diffusion e a criação de modelos 3D. O curso de Stable Diffusion mostram diferentes técnicas para gerar imagens com maior controle, especialmente utilizando ControlNet e LoRA. Essas técnicas podem auxiliar na geração de ideias visuais de personagens, ambientes e objetos para o desenvolvimento de jogos. Além disso, outras publicações detalham os passos para gerar texturas e efeitos de paralaxe com Stable Diffusion. Já os fluxos de trabalho para criação de modelos 3D utilizam plataformas de IA gerativa, como CSM, Meshy AI, Luma Genie AI e Kaedim, que apresentam resultados satisfatórios e eficientes nos projetos relatados pelos autores.

A análise da revisão sistemática realizada revelou tendências e padrões claros no uso da IA gerativa na modelagem 3D para jogos. A maior parte das publicações recentes (2023-2024) concentra-se na geração de modelos 3D, demonstrando o interesse crescente na automação e otimização do processo de criação de personagens e objetos. A geração de texturas e materiais PBR também é significativa, evidenciando a preocupação com a consistência visual e a qualidade estética dos ativos digitais. Já o auxílio ao fluxo de trabalho, como geração de conceitos visuais, mapas de profundidade e efeitos de paralaxe, embora menos explorado, mostra potencial para otimizar etapas intermediárias do *pipeline* de criação.

Para complementar a análise, a Tabela 1 as tecnologias utilizadas e suas aplicações na modelagem 3D, destacando suas aplicações em geração de modelos, texturização e suporte ao fluxo de trabalho. Observa-se que a maior parte das ferramentas foca na criação de modelos 3D, enquanto algumas abordam a texturização e/ou fluxo de trabalho.

Tabela 1 – Resumo de aplicações de ferramentas

<b>Tecnologia / Artigo</b>	<b>Aplicação</b>
MVDreams	Modelos 3D
One-2-3-45++	Modelos 3D
CSM.ai	Modelos 3D
Meshy AI	Modelos 3D
Luma Genie AI	Modelos 3D
Kaedim	Modelos 3D
GenesisTex	Texturização
Material Palette	Texturização
Stable Diffusion + ControlNet/LoRA	Fluxo de trabalho
3D Parallax	Fluxo de trabalho / Texturização

Além disso, a revisão sistemática identificou alguns desafios recorrentes: a dificuldade de controle preciso sobre as gerações, o elevado custo computacional, a necessidade de pós-processamento para garantir coerência e consistência, e o risco de resultados semelhantes aos dados utilizados no treinamento. Esses desafios reforçam a importância de guias e manuais práticos que orientem o uso eficiente das ferramentas, proporcionando maior controle aos desenvolvedores, especialmente aqueles sem experiência prévia em modelagem 3D ou habilidades artísticas.

Em síntese, os resultados da revisão sistemática confirmam que a IA gerativa possui potencial significativo para acelerar a produção de modelos 3D, texturas e ideias visuais, aumentando a produtividade e democratizando o acesso a técnicas avançadas de criação digital. Estes fundamentam a etapa subsequente deste trabalho: o desenvolvimento de um manual prático detalhado para o uso de IA gerativa no desenvolvimento de jogos.

### 3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, são abordados conceitos, tipos e abordagens de IA gerativa fornecendo uma base sólida para compreender o papel dessa ferramenta para criação de ideias e texturas na modelagem 3D.

#### 3.1 DEFINIÇÃO E CONCEITOS DE IA GERATIVA

A IA gerativa refere-se a uma categoria de modelos e ferramentas de inteligência artificial projetadas para criar novos conteúdos, como texto, imagens, vídeos, músicas ou código. Essa tecnologia utiliza diferentes técnicas, incluindo redes neurais e algoritmos de aprendizado profundo (*Deep Learning*) para identificar padrões e gerar novos resultados. Trata-se de um subcampo da inteligência artificial focado na criação de conteúdos a partir de conjuntos de dados existentes. Esses algoritmos de IA aprendem com os dados fornecidos e são capazes de gerar saídas semelhantes, mas não idênticas, com base no conhecimento adquirido durante o treinamento (Data Science Academy, 2023). O termo “gerativa” deriva do verbo “gerar”, indicando a capacidade dessas tecnologias de criar novos conteúdos a partir de informações prévias.

#### 3.2 CONTEXTO HISTÓRICO

As origens da IA remontam à década de 1950 e ao matemático e cientista da computação britânico Alan Turing. Em sua proposta conhecida como Teste de Turing mede a capacidade de uma máquina exibir um comportamento inteligente indistinguível do de um ser humano (Rudina Seseri; Dolce, 2024). Embora sua proposta estivesse mais voltada à questão da inteligência de forma geral, essa proposta lançou as bases filosóficas e técnicas para o surgimento da IA, influenciando diretamente os estudos subsequentes sobre como máquinas poderiam simular processos cognitivos humanos.

Outro marco importante na mesma década ocorreu em 1956, com a criação do programa *Logic Theorist* por Allen Newell, Cliff Shaw e Herbert Simon, considerado o primeiro sistema de IA da história. Financiado pela RAND Corporation, o programa foi projetado para simular a resolução de problemas matemáticos, imitando o raciocínio humano (Alalaq, 2025). Nesse mesmo ano, o matemático John McCarthy organizou a Conferência de Dartmouth, reunindo pesquisadores de diversas áreas para discutir a possibilidade de construir máquinas inteligentes. Foi nesse evento que McCarthy cunhou o termo "inteligência artificial", definindo a área de pesquisa como o esforço de criar sistemas capazes de executar tarefas que exigem inteligência humana (Alalaq, 2025). Em 1957, Frank Rosenblatt desenvolveu o Perceptron, o primeiro algoritmo de aprendizado supervisionado para classificadores binários, estabelecendo as bases para as redes neurais e o aprendizado de máquina inicial (Singh, 2024).

Na década seguinte, um avanço notável ocorreu com o desenvolvimento do programa ELIZA, criado por Joseph Weizenbaum em 1966. Esse programa pioneiro foi desenvolvido para simular uma sessão de terapia reutilizando as respostas fornecidas pelo próprio usuário para formular novas perguntas e assim estimular a continuidade da conversa. Weizenbaum tinha a intenção de demonstrar a simplicidade da inteligência das máquinas ao evidenciar as limitações do ELIZA. No entanto, a capacidade do chatbot de engajar os usuários em conversas foi surpreendentemente eficaz, e muitas pessoas acreditavam estar interagindo com um terapeuta humano (Alalaq, 2025). Assim, o ELIZA representou uma das primeiras demonstrações práticas de geração automatizada de texto em resposta à entrada do usuário, contribuindo significativamente para os fundamentos conceituais e técnicos dos sistemas modernos de IA gerativa em linguagem natural.

Publicado em 1969, *Perceptrons: An Introduction to Computational Geometry*, de Minsky e Papert, é uma obra fundamental na história da IA que analisou as capacidades e limitações dos perceptrons, um tipo inicial de neurônio artificial. O livro destacou que os perceptrons não conseguiam processar certos tipos de dados, incluindo aqueles que não são linearmente separáveis, o que levou a uma redução significativa no interesse e no financiamento para pesquisas com redes neurais (Rudina Seseri; Dolce, 2024). Esse cenário contribuiu para o que ficou conhecido como os “invernos da IA” nas décadas de 1970 e 1980 — períodos marcados por redução de recursos financeiros, desaceleração das pesquisas e frustração com os resultados práticos obtidos. Esse movimento foi essencial para a evolução do aprendizado profundo e para o ressurgimento da IA gerativa baseada em redes neurais nas décadas seguintes. Em 1973, Harold Cohen desenvolveu o AARON, um dos primeiros programas capazes de gerar arte de forma autônoma, demonstrando as primeiras formas de IA gerativa (Singh, 2024). Embora não utilizasse redes neurais ou aprendizado de máquina, o AARON foi um dos primeiros exemplos de uma IA gerativa capaz de produzir resultados criativos.

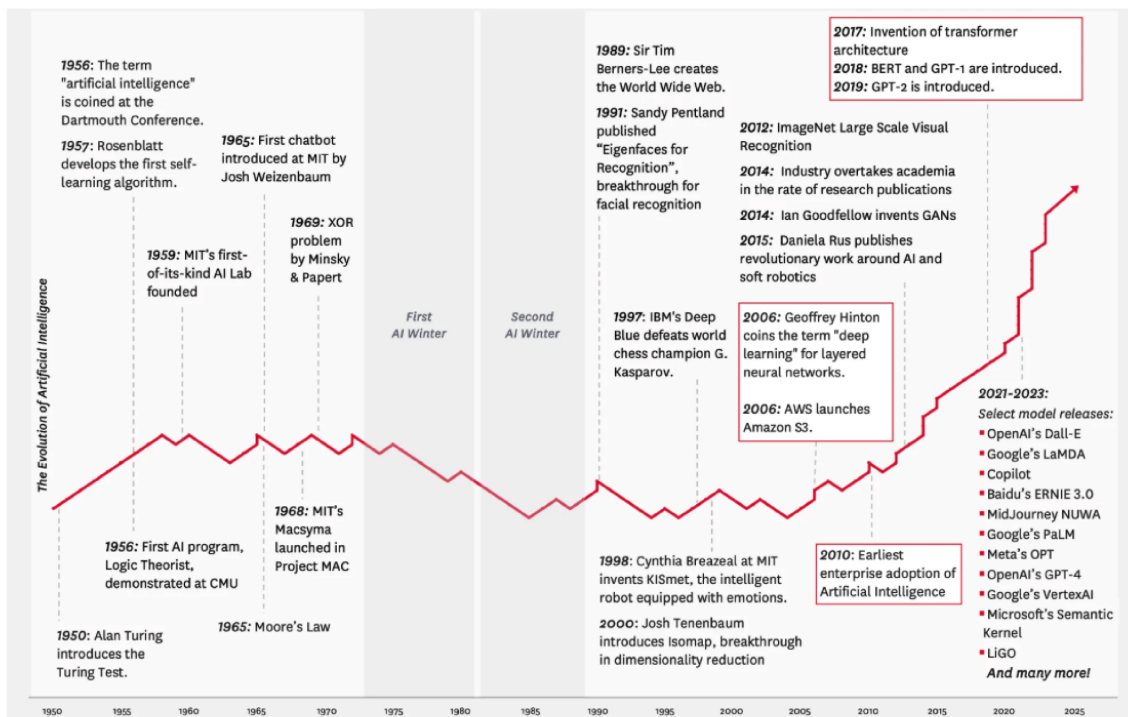
Em seguida, nas décadas de 1980 e 1990, apesar das limitações enfrentadas durante os chamados “invernos da IA”, importantes avanços continuaram a ser feitos no campo da IA. Em 1986, Geoffrey Hinton, David Rumelhart e Ronald J. Williams popularizaram o algoritmo de retropropagação (*backpropagation*), permitindo o treinamento eficiente de redes neurais multicamadas (Singh, 2024). Este foi um avanço fundamental na IA e levou ao renascimento do interesse em redes neurais. Esses modelos introduziram a ideia de aprender representações internas dos dados, que poderiam ser utilizadas para gerar novas amostras de forma criativa.

Nos anos seguintes, textos e estudos aprofundados passaram a se destacar em mecanismos de busca e acervos bibliográficos, quando Tom Mitchell publicou seu livro *Machine Learning* em 1997. Pela primeira vez, Mitchell definiu formalmente o aprendizado de máquina como “um programa de computador que aprende com a experiência”. Com base nesse conceito, um grupo de pesquisadores aprofundou e desenvolveu a ideia em

seu estudo de aprendizado profundo (Alalag, 2025). Na década de 2000, o crescimento da capacidade computacional e a disponibilidade de grandes bases de dados viabilizaram experimentos mais ambiciosos com redes neurais profundas, dando origem a modelos mais sofisticados de geração de dados.

Um marco significativo na história da IA gerativa ocorreu em 2014, com a introdução das *Generative Adversarial Network* (GANs) por Ian Goodfellow e seus colegas, que demonstraram ser uma abordagem eficaz para gerar conteúdo novo e realista. Essa técnica baseia-se em uma estrutura onde duas redes neurais (gerador e discriminador) competem para gerar dados realistas. As GANs se tornaram essenciais na área de IA gerativa, usada para criar imagens e vídeos de alta qualidade (Singh, 2024). Elas tiveram um impacto profundo em campos como visão computacional, processamento de linguagem natural e arte gerativa. Simultaneamente, os *Variational Autoencoder* (VAE) emergiram como uma técnica poderosa para geração de dados. Nos anos seguintes, a IA gerativa continuou a evoluir com o surgimento de novas técnicas e modelos, como Transformers e GPT. Em 2021, foram lançadas ferramentas de IA gerativa baseadas em modelos de difusão como DALL-E, Stable Diffusion e Midjourney que marcaram o início de uma nova era de criatividade e inovação tecnológica. Esses modelos permitem gerar imagens de alta qualidade a partir de descrições textuais. Uma linha do tempo representativa é mostrada na Figura 1, traçando a trajetória de desenvolvimento dos métodos e aplicações da IA.

Figura 1 – Linha do tempo da história de IA.



Fonte: Rudina Seseri e Dolce (2024)

### 3.3 PRINCIPAIS ABORDAGENS DE IA GERATIVA

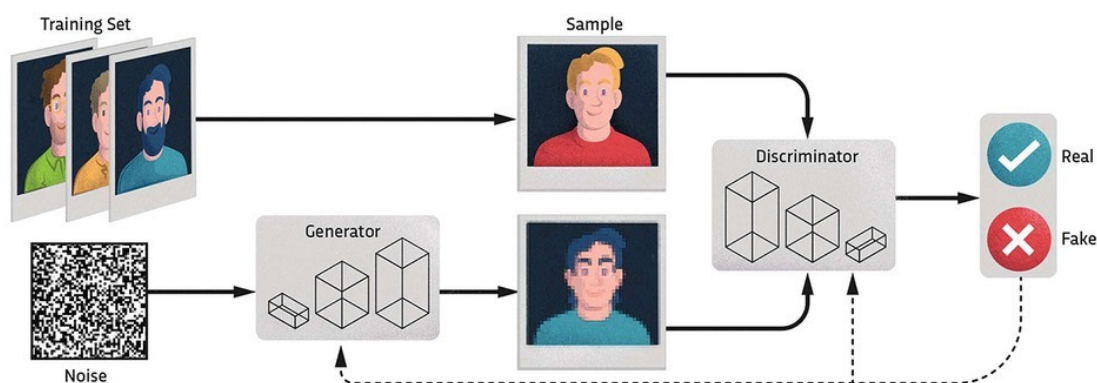
Nesta seção, será apresentado o conceito básico sobre duas das principais abordagens em IA gerativa: as GANs e os modelos de difusão. Essas técnicas têm revolucionado a maneira como se criam e se manipulam dados, oferecendo novas possibilidades em diversos campos.

#### 3.3.1 Redes Adversariais Generativas (GANs)

Uma das abordagens mais populares para a síntese de imagens é utilizar um tipo de GAN projetada para aprender um domínio específico de imagens e, em seguida, gerar imagens totalmente novas, ou seja, os dados sintéticos (Kováčik, 2024). A implementação envolve dois modelos que competem entre si: um modelo gerador e um modelo discriminador. O modelo discriminador é uma rede neural que tenta distinguir os dados reais dos dados criados pelo gerador, enquanto o modelo gerador é outra rede neural que cria amostras que são indistinguíveis dos dados reais, enganando assim o discriminador (D.Google, 2022).

Durante o treinamento, o gerador recebe uma entrada aleatória, geralmente na forma do ruído, e utiliza técnicas de aprendizado para transformá-la em uma instância de dados sintéticos. Além disso, para melhorar a qualidade dos dados gerados, o gerador incorpora o feedback fornecido pelo discriminador, ajustando seus parâmetros para produzir dados que enganam o discriminador (Kováčik, 2024). Por outro lado, o discriminador recebe instâncias de dados reais, provenientes de conjuntos de dados autênticos e instâncias de dados falsos, geradas pelo gerador. Para melhorar, o discriminador aprende sendo penalizado por classificação errada. A representação visual da arquitetura GANs pode ser visualizada na Figura 2.

Figura 2 – Arquitetura de GANs.



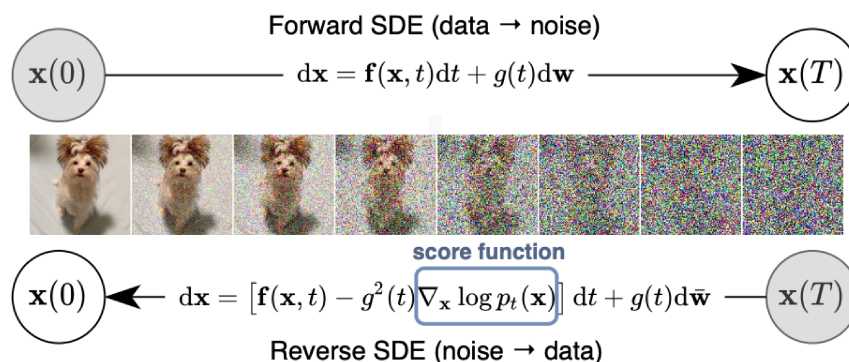
### 3.3.2 Modelos de Difusão

Os modelos de difusão se destacaram como uma inovação significativa no campo do aprendizado de máquina, particularmente em tarefas de geração e restauração de dados. Eles são considerados uma classe de modelos de IA gerativa capazes de gerar imagens de alta qualidade, ao contrário de outros modelos, como GANs e VAEs, que apresentam dificuldades para produzir imagens detalhadas em alta resolução (Ahirwar, 2023). A essência do modelo de difusão reside em dois processos fundamentais: o processo de adição e remoção de ruído.

O primeiro passo no funcionamento de um modelo de difusão é o processo de adição de ruído, também conhecido como processo direto. Nesse estágio, dados não estruturados, como imagens claras e nítidas, são gradualmente transformados em dados ruidosos. O processo começa com a adição de uma pequena quantidade de ruído gaussiano à imagem original, repetida em várias etapas, aumentando progressivamente o ruído até que a imagem se transforme quase completamente em ruído puro.

Em seguida, após concluir o processo de adição de ruído, inicia-se o processo de remoção de ruído, com o objetivo de reverter o processo anterior, eliminando o ruído e recuperando a imagem original (Ahirwar, 2023). Para isso, utiliza-se um modelo de aprendizado profundo, geralmente uma rede neural convolucional, eficiente na captura de padrões espaciais. A partir do treinamento, o modelo aprende a recuperar os dados originais a partir do ruído, utilizando o caminho inverso da cadeia de Markov, mapeando a distribuição complexa dos dados de volta para uma distribuição simples, permitindo que o espaço latente represente recursos significativos, padrões e variáveis latentes presentes nos dados (Walker, 2023). Assim, após o processo de treinamento, o modelo de difusão é capaz de realizar tarefas notáveis, como gerar novas imagens ou restaurar imagens degradadas a partir de entradas ruidosas, como apresentado na Figura 3.

Figura 3 – Passos para adição e remoção de ruído.



Ao contrário dos modelos gerativos, como as GANs ou os VAEs, que geram imagens por meio de uma única passagem direta, os modelos de difusão exigem múltiplas passagens sucessivas. Essa característica impõe uma carga computacional mais elevada durante o treinamento, pois o modelo deve aprender a remoção de ruído em várias escalas. Além disso, a natureza iterativa do processo aumenta o tempo de inferência, tornando os modelos de difusão computacionalmente menos eficientes do que outros modelos gerativos (Tero Karras Miika Aittala; Laine, 2022).

## 3.4 TECNOLOGIAS DA IA GERATIVA

Esta seção apresenta os principais conceitos e tecnologias empregadas no contexto deste trabalho, incluindo o modelo Stable Diffusion e seus diferentes modos de uso, como *text-to-image* e *image-to-image*, além de extensões e técnicas complementares, como o ControlNet e o LoRA. A compreensão dessas tecnologias é fundamental para contextualizar o desenvolvimento do manual prático e sua aplicação na geração de modelos 3D.

### 3.4.1 Stable Diffusion

O Stable Diffusion é um grande modelo de difusão para geração de imagens a partir do texto, treinado com bilhões de imagens. Ele foi desenvolvido pela empresa Stability AI, em parceria com pesquisadores da CompVis (Universidade de Heidelberg) e da RunwayML, e lançado em 2022 como um modelo de código aberto. O Stable Diffusion utiliza representações latentes codificadas a partir dos dados de treinamento como entrada (Mishra, 2023).

A geração de imagens pelo modelo é realizada por meio da codificação dos *prompts* em vetores latentes, utilizando modelos de linguagem pré-treinados, e pela aplicação do processo de difusão nesse espaço comprimido. O uso do espaço latente reduz o consumo de memória e o tempo computacional em comparação ao processamento direto no espaço de *pixels*, o que facilita tanto o treinamento quanto a inferência do modelo (Rombach et al., 2022).

Além disso, por ser código aberto e apresentar alta eficiência, o Stable Diffusion consolidou-se como uma ferramenta versátil para a geração rápida de imagens de alta qualidade, sendo amplamente utilizado em fluxos de criação artística.

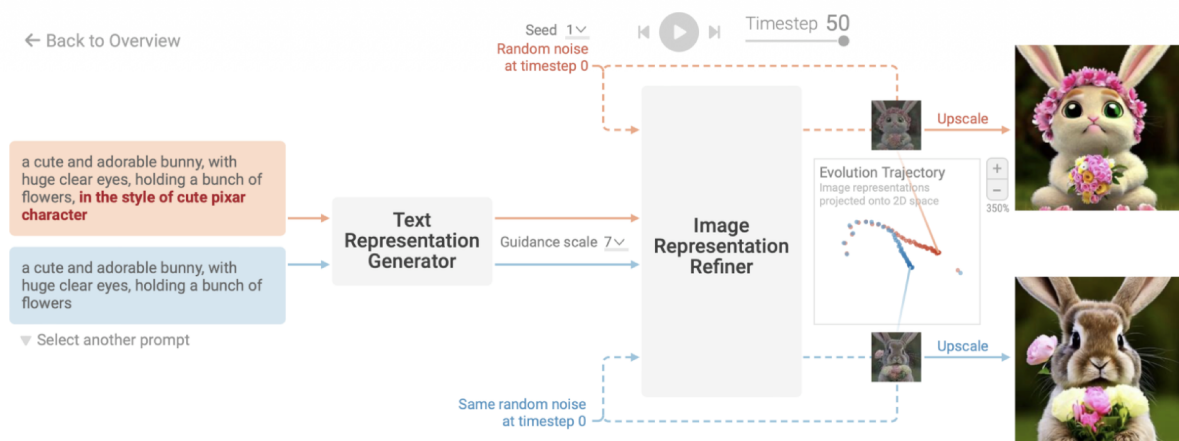
### 3.4.2 Text-to-Image e Image-to-Image

O *text-to-image* é uma técnica de geração de imagens a partir de um *prompt* de texto. O processo inicia-se com a tokenização do texto, em que a descrição é dividida em unidades linguísticas menores. Essas unidades são então codificadas em vetores latentes por meio de modelos como o *CLIP Text Encoder* (Lee et al., 2023). Em seguida, esses vetores guiam a criação da imagem: o processo começa a partir de uma amostra inicialmente ruidosa e,

por meio de um procedimento iterativo de redução de ruído — denominado *denoising* —, a imagem final é construída, aproximando-se da representação visual correspondente ao prompt textual.

A Figura 4 ilustra como a escolha de palavras em um prompt textual pode influenciar significativamente a geração da imagem no modelo Stable Diffusion. Essa visualização destaca a importância da escolha precisa das palavras nos *prompts* para controlar aspectos específicos da imagem gerada, como estilo, composição e elementos visuais. Além disso, demonstra como o modelo interpreta e responde a modificações sutis na descrição textual, proporcionando uma compreensão mais profunda do processo de geração de imagens.

Figura 4 – Exemplo de text-to-image.



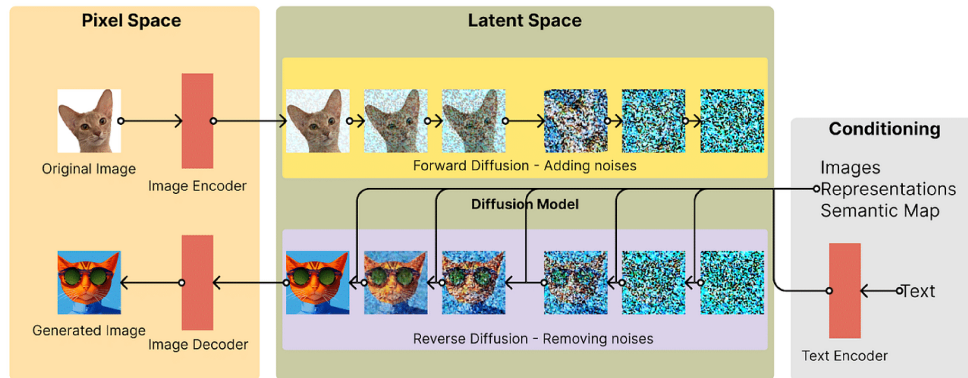
Fonte: LEE, Seongmin et al. Diffusion Explainer: Visual Explanation for Text-to-image Stable Diffusion (2024). Disponível em: <https://arxiv.org/pdf/2305.03509>

A técnica de image-to-image é uma extensão do text-to-image, na qual o modelo recebe como entrada uma imagem inicial, transformando-a em uma nova imagem, mantendo os elementos estruturais e o conceito original. Portanto, o image-to-image permite refinar, estilizar ou alterar a imagem de entrada, com o prompt textual como guia.

Dessa forma, o image-to-image combina a interpretação semântica do texto com a preservação das características visuais da imagem de entrada para gerar uma nova imagem. O processo de geração é semelhante ao text-to-image, mas, em vez de começar com uma imagem totalmente ruidosa, ele utiliza a imagem de entrada como base para o refinamento.

A Figura 5 ilustra a arquitetura geral do modelo Stable Diffusion. A imagem inicial é primeiro codificada em um espaço latente. Em seguida, realiza o processo de denoising, no qual o ruído é removido iterativamente da representação latente. Durante esse processo, o prompt textual serve para guiar a geração da imagem. Por fim, a representação latente é decodificada de volta para o espaço de pixels, produzindo a imagem final.

Figura 5 – Exemplo de image-to-image.



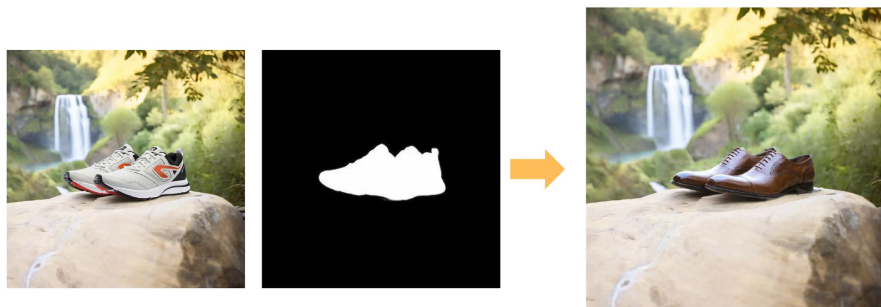
Fonte: NGUYEN, Sang. Stable Diffusion: The Expert Guide (2025). Disponível em: <https://bestarion.com/stable-diffusion-the-expert-guide/>

### 3.4.3 Inpainting

O inpainting é uma técnica de edição de imagem que permite preencher, apagar ou substituir partes de uma imagem. No Stable Diffusion, o inpainting é realizado usando uma máscara que indica a região a ser modificada. Um prompt textual pode ser fornecido para orientar como a região mascarada deve ser modificada. Além disso, o processo é similar ao text-to-image e image-to-image, o modelo aplica denoising iterativo sobre a área mascarada, preservando o restante da imagem sem alterações indesejadas.

A Figura 6 mostra o inpainting no Stable Diffusion. Uma máscara foi aplicada sobre o tênis, e o modelo, guiado por um prompt textual, substituiu o tênis por um sapato social, mantendo o fundo da natureza inalterado.

Figura 6 – Exemplo de inpainting.



Fonte: KARTHIK, Shanmukha. In-Depth Guide to Stable Diffusion Inpainting Techniques (2024).

### 3.4.4 ControlNet

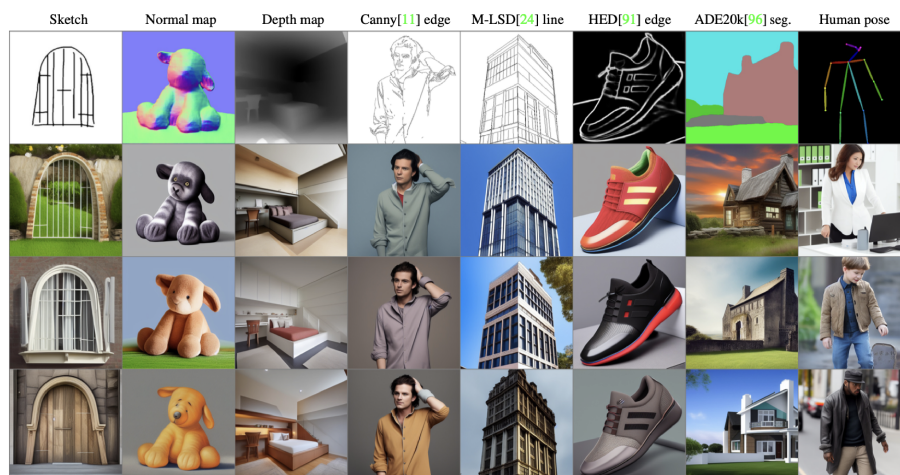
O ControlNet é uma arquitetura de rede neural projetada para adicionar controles condicionais espaciais a modelos de difusão. Ele permite que os usuários forneçam condições adicionais, como esboços, mapas de bordas, mapas de profundidade ou poses humanas, para guiar a geração da imagem com maior precisão. Além disso, o ControlNet cria uma cópia treinável das camadas do modelo principal, conectadas por camadas convolucionais inicializadas com zeros, garantindo que o modelo original não seja alterado durante o treinamento (Zhang; Rao; Agrawala, 2023).

Além disso, o ControlNet permite controlar o processo criativo por meio de diferentes tipos de parâmetros. Esses controles permitem que o usuário direcione o modelo de IA para gerar imagens mais próximas de suas intenções, garantindo consistência e precisão nos detalhes desejados. Entre os tipos de controle mais comuns, destacam-se:

- **Canny Edge:** guia o modelo pelas bordas da imagem de referência, mantendo a forma dos objetos.
- **Open Pose:** define a pose de personagens, útil para gerar referências de postura antes da modelagem 3D.
- **Depth/Normal Map:** fornece informações de profundidade, facilitando a percepção tridimensional para modelagem.

A Figura 7 apresenta exemplos de controles para geração de imagens. Essa figura evidencia como o ControlNet permite maior controle sobre a posição, forma e elementos visuais das imagens geradas, combinando informações estruturais fornecidas pelo usuário com orientação textual e mantendo coerência e detalhes na saída final.

Figura 7 – Exemplos de ControlNet.



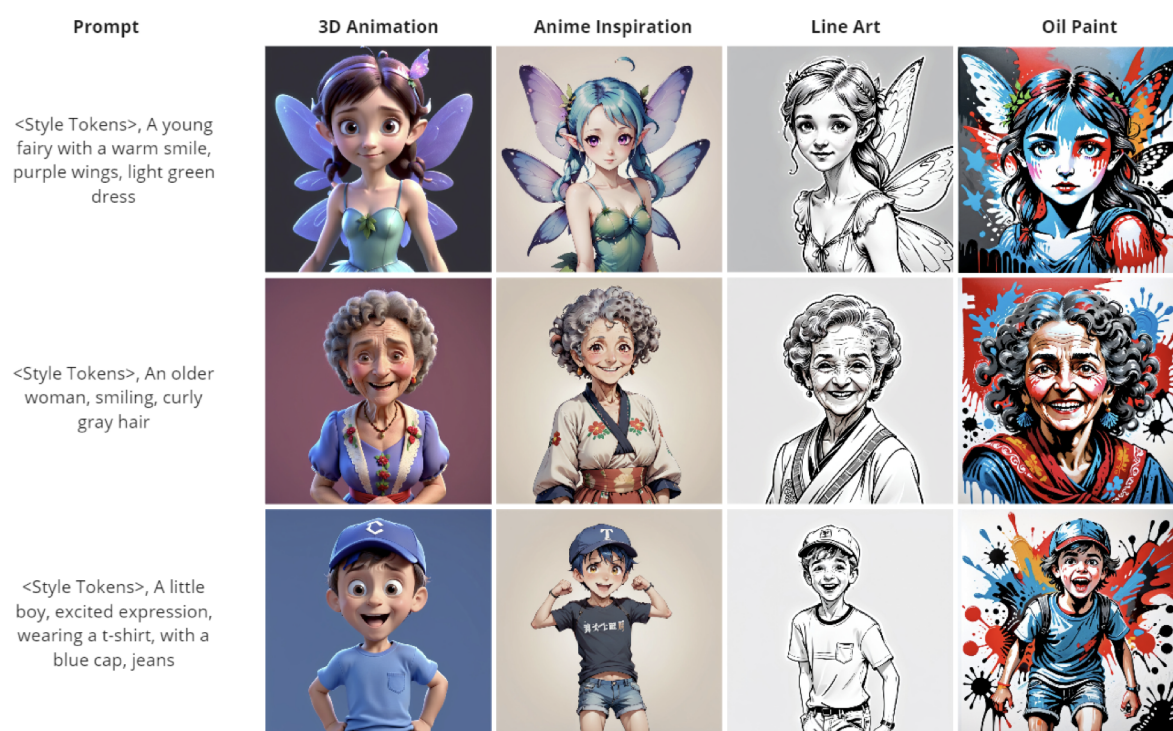
### 3.4.5 LoRA

O *Low-Rank Adaptation* (LoRA) é uma técnica que permite adaptar modelos de aprendizado profundo de grande escala com baixo custo computacional. Em vez de treinar todos os pesos do modelo, o LoRA insere pequenas matrizes de baixo *rank* nas camadas existentes, ajustando apenas essas matrizes durante o treinamento. Dessa forma, é possível ensinar o modelo a gerar imagens ou estilos específicos sem alterar os pesos originais, reduzindo significativamente o tempo, a memória e os recursos de processamento (Hu et al., 2022).

Essa abordagem é útil na criação de imagens e personagens para jogos, pois o LoRA pode ser treinado com um estilo visual específico ou até mesmo para reproduzir um personagem particular.

A Figura 8 ilustra exemplos de imagens geradas com LoRAs treinados com estilos distintos. É possível observar que o mesmo prompt pode gerar estilos diferentes, dependendo do LoRA utilizado.

Figura 8 – Exemplos de LoRA.



Fonte: CUI, Zhipu et al. AC-LoRA: Auto Component LoRA for Personalized Artistic Style Image Generation (2025). Disponível em: <https://arxiv.org/pdf/2504.02231>

### 3.5 FERRAMENTAS UTILIZADAS

Diversas interfaces e plataformas foram desenvolvidas para tornar mais acessível a manipulação de modelos de IA generativa por usuários técnicos e não técnicos. Essas ferramentas apresentam perfis distintos, variando desde interfaces minimalistas voltadas a iniciantes até soluções mais robustas, otimizadas para produção e colaboração. Para contextualizar a escolha das ferramentas adotadas neste trabalho, a Tabela 2 apresenta uma comparação entre algumas das principais interfaces WebUI para Stable Diffusion. Observa-se que, embora existam alternativas mais simples ou mais especializadas, o Automatic1111 e o ComfyUI se destacam por oferecerem um equilíbrio superior entre flexibilidade, recursos avançados, suporte a extensões e ampla adoção pela comunidade.

Tabela 2 – Comparação entre principais interfaces WebUI para Stable Diffusion

WebUI	Uso recomendado	Dificuldade	Diferencial
Automatic1111	Usuários em geral	Moderada	Muitas extensões
ComfyUI	Usuários avançados	Alta	Fluxo baseado em nós
Foocus	Iniciantes	Baixa	Simplicidade
Forge	Desempenho	Moderada	Otimização de velocidade
EasyDiffusion	Uso imediato	Muito baixa	Instalação 1-clique
InvokeAI	Profissionais	Moderada	Editor de canvas
SD.Next	Usuários avançados	Alta	Recursos mais recentes
SwarmUI	Equipes	Moderada	Multiusuário
MetaStable	Uso simplificado	Baixa	Interface limpa
StabilityMatrix	Múltiplas instalações	Baixa	Gerenciador de WebUI
MochiDiffusion	Usuários Mac	Fácil	Nativo no macOS

Fonte: Adaptado de Propel RC (2025). Disponível em:  
<https://www.propelrc.com/11-best-stable-diffusion-webuis>.

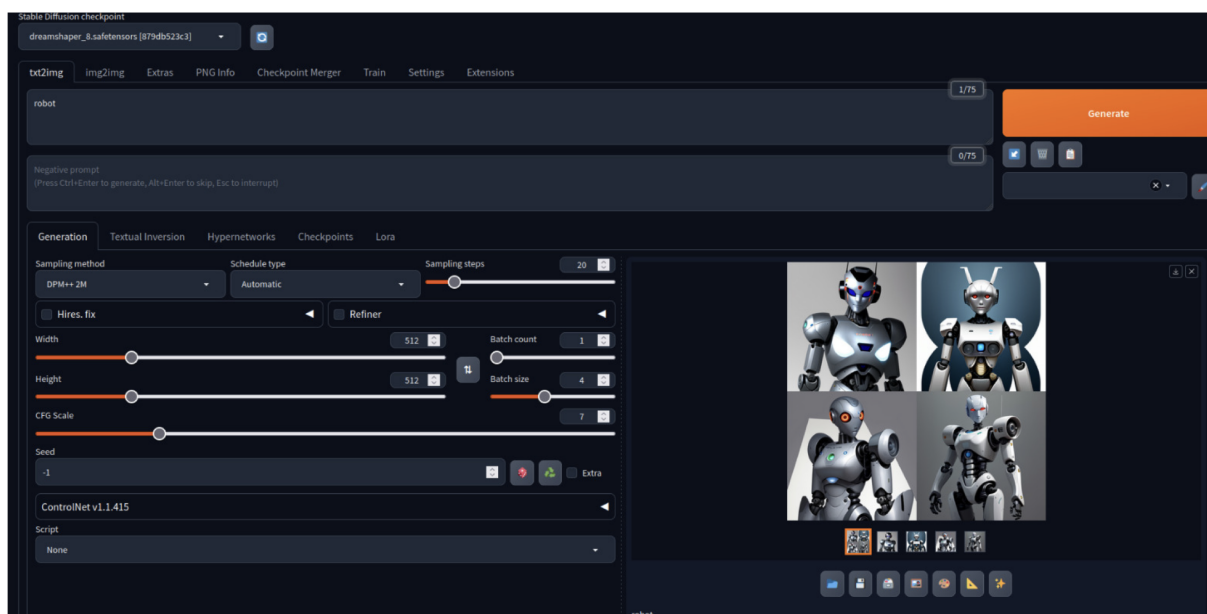
Esta seção apresenta as ferramentas exploradas neste trabalho, destacando suas características, motivações para a escolha e contribuições específicas na geração de imagens e modelos aplicados ao desenvolvimento de jogos.

#### 3.5.1 Automatic1111

O Automatic1111 é uma das primeiras interfaces gráficas baseadas na web a permitir a utilização do modelo Stable Diffusion de forma interativa. Trata-se de uma ferramenta de código aberto que oferece um ambiente coeso e intuitivo, reunindo em uma única interface todos os recursos essenciais para a criação de imagens geradas por IA. Entre suas funcionalidades, destacam-se o suporte a ControlNet, LoRA, o ajuste de *prompts* e diversas extensões desenvolvidas pela comunidade. Essa flexibilidade torna o Automatic1111 uma ferramenta popular tanto para usuários iniciantes quanto para avançados, que desejam explorar a geração de imagens personalizadas.

A Figura 9 ilustra a interface do Automatic1111, evidenciando a disposição dos principais painéis de controle e os parâmetros de geração de imagens utilizados pelo modelo Stable Diffusion.

Figura 9 – Interface de Automatic1111.

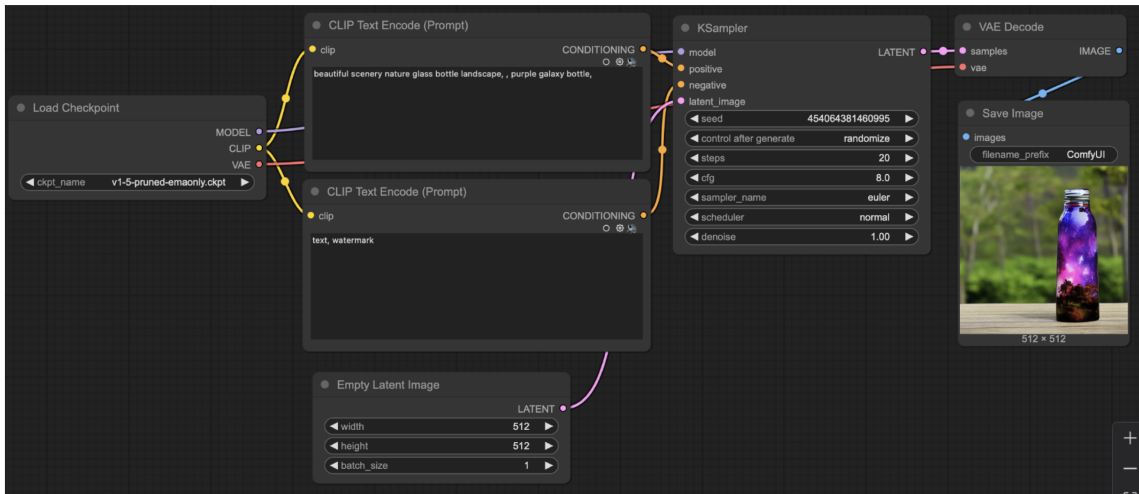


### 3.5.2 ComfyUI

O ComfyUI se tornou uma das interfaces web de código aberto que mais cresce na comunidade do Stable Diffusion, destacando-se por ser uma alternativa mais poderosa e flexível em comparação a outras plataformas. Diferentemente do Automatic1111, o ComfyUI adota uma abordagem visual baseada em nós, na qual o usuário define o fluxo de trabalho conectando módulos entre si, de forma semelhante a um fluxograma. Essa estrutura permite maior controle sobre cada etapa do processo de geração, facilitando a personalização, a integração de extensões como ControlNet e LoRA, e a criação de fluxos de trabalho complexos para diferentes estilos e aplicações.

A Figura 10 apresenta a interface do ComfyUI, é possível observar o ambiente de criação visual baseado em nós, no qual o usuário organiza o *workflow* (fluxo de trabalho) conectando diferentes módulos de processamento de imagem.

Figura 10 – Interface de ComfyUI.



A Tabela 3 sintetiza os principais aspectos comparativos entre Automatic1111 e ComfyUI. Automatic1111 se mostra mais apropriada para quem busca rápida familiarização e geração pontual de imagens – ideal para prototipagem, experimentação ou uso eventual. Por outro lado, o ComfyUI oferece vantagens importantes para tarefas que exigem fluxos de trabalho mais complexos, como geração de vídeos, criação de modelos 3D e automação de pipelines de produção, recursos que não estão disponíveis no Automatic1111. Por essas características, o ComfyUI é ideal para etapas que demandam consistência, reutilização e maior controle sobre o processo de criação de ativos digitais.

Tabela 3 – Comparativo entre Automatic1111 e ComfyUI.

<b>Critério</b>	<b>Automatic1111</b>	<b>ComfyUI</b>
Instalação	Instalação com 1-clique, simples e rápida	Requer setup manual (dependências, configurações)
Curva de aprendizado	Fácil de usar – ideal para iniciantes	Mais complexa para iniciantes – exige tempo para entender baseada em nós
Rapidez em tarefas simples	Rápido para imagens isoladas e simples	Leve vantagem ou equivalente (fluxos simples)
Desempenho em workflows complexos	Menos eficiente, uso de <i>Video Random Access Memory</i> (VRAM) maior	30–60% mais rápido, melhor uso de VRAM, viável em hardware modesto
Reutilização	Parâmetros devem ser manualmente definidos a cada geração	Workflows salváveis (no arquivo JSON), consistência entre execuções
Flexibilidade	Extensões amplas e comunidade madura	Suporte rápido a novos modelos, pipelines complexos, mistura de técnicas
Perfil ideal de uso	Geração rápida, experimentação, uso pontual	Produção profissional, fluxos complexos e suporte a técnicas avançadas (vídeo, áudio e modelos 3D)

Fonte: Adaptado de Apatero (2025). Disponível em: <https://apatero.com/blog/comfyui-vs-automatic1111-which-should-you-use-2025>.

## 4 DESENVOLVIMENTO DO MANUAL

Nesta seção, apresenta-se o desenvolvimento de um manual prático (Kiba; Wangenheim, 2025) — disponível no Apêndice B — para a utilização de ferramentas de IA gerativa na criação de modelos 3D, especialmente personagens, destinados a jogos 3D desenvolvidos na Unreal Engine 5. O manual foi elaborado com foco em usuários com pouca experiência em modelagem, bem como em estudantes de computação gráfica, oferecendo instruções que permitem criar modelos 3D de forma criativa e ágil.

O manual está organizado de forma a conduzir o usuário progressivamente, iniciando com a apresentação das ferramentas utilizadas, passando pela geração de imagens e personagens, e concluindo com a integração dos ativos gerados à Unreal Engine 5. Cada capítulo foi planejado para fornecer instruções detalhadas, claras e ilustradas, combinando explicações e exemplos práticos visuais, permitindo que o usuário compreenda tanto os conceitos envolvidos quanto sua aplicação direta.

A seguir, cada etapa do manual será resumida, mostrando como os conceitos teóricos podem ser aplicados na prática, desde a criação de imagens de referência até a implementação de personagem no jogo.

### 4.1 GERAÇÃO DE IMAGEM

A base para a criação dos modelos 3D é a geração de imagens, que é detalhada utilizando as interfaces Automatic1111 e ComfyUI. O manual apresenta instruções sobre como utilizar ambas as interfaces, ensinando técnicas de text-to-image e image-to-image.

Durante a geração de imagem, diversos parâmetros podem influenciar o resultado visual. O usuário pode ajustar esses parâmetros, testando e personalizando-os para gerar a imagem desejada, observando como cada configuração influencia o resultado final. O manual descreve os principais parâmetros:

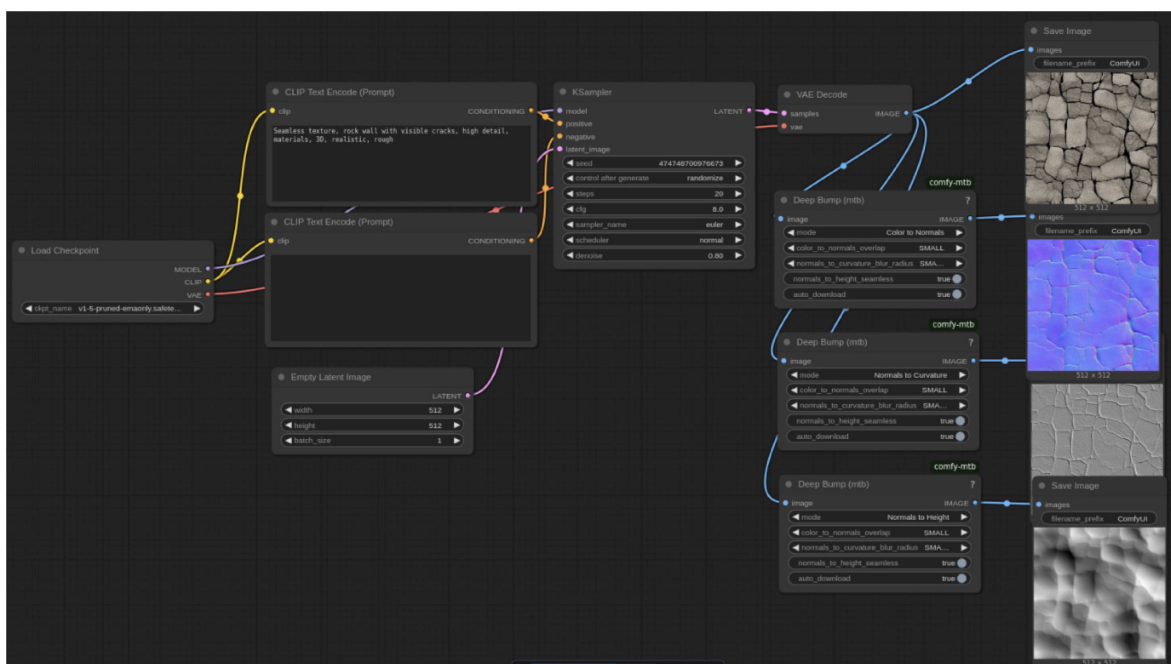
- **Prompt Positivo:** descreve os elementos que devem estar presentes na imagem.
- **Prompt Negativo:** define o que deve ser evitado na imagem, ajudando a refinar o resultado.
- **Peso das Palavras-chave:** controla a importância de termos específicos dentro do prompt, atribuindo pesos.
- **Steps (passos):** número de iterações do modelo, geralmente, maior quantidade de steps aumenta o detalhamento.
- **CFG Scale:** ajusta o quanto o modelo segue o prompt, equilibrando fidelidade e criatividade.
- **Sampler:** algoritmo usado para remoção de ruído, influenciando estilo e textura.

- **Modelos de Difusão:** escolha do modelo base, que determina o estilo e a qualidade da imagem gerada.
- **Batch Count/Batch Size:** Batch Size determina o número de imagens geradas simultaneamente em cada iteração, enquanto o Batch Count indica quantas vezes esse lote será processado.
- **Denoise:** determina a quantidade de ruído adicionada à imagem inicial antes do processo de geração.

Além disso, o manual dedica uma subseção à geração de mapas de texturas essenciais para materiais 3D realistas, como *Base Color*, *Normal Map*, *Height Map* e *Curvature Map*, detalhando como automatizar essa conversão usando nós customizados no ComfyUI (Figura 11). O *Base Color* define as cores básicas da superfície, enquanto o *Normal Map* adiciona detalhes de relevo sem alterar a geometria do modelo. O *Height Map* representa a variação de altura da superfície, permitindo efeitos de profundidade adicionais, e o *Curvature Map* evidencia a curvatura e contornos do objeto, útil para sombreado e aplicação de efeitos de desgaste.

Ao automatizar a geração desses mapas com nós no ComfyUI, o manual oferece aos usuários a possibilidade de produzir rapidamente texturas detalhadas e consistentes, mesmo sem amplo conhecimento em modelagem 3D ou texturização manual. Essa metodologia permite reduzir o esforço e o tempo necessários para a produção de texturas, garantindo maior previsibilidade e fidelidade visual dos ativos.

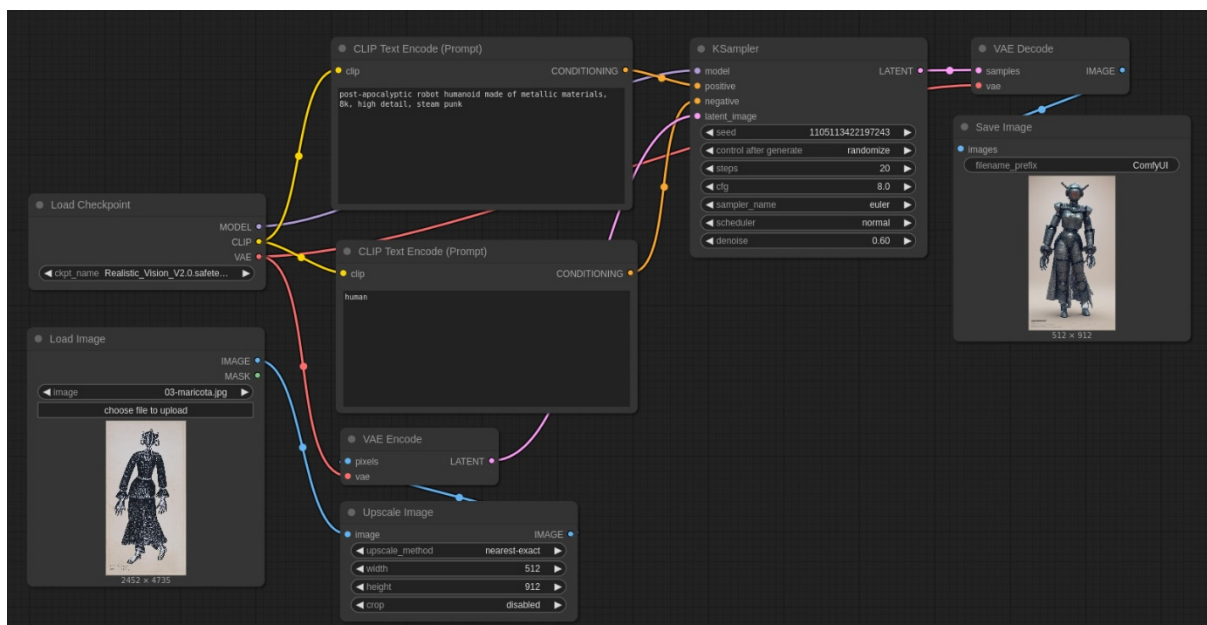
Figura 11 – Exemplo de geração de textura - ComfyUI.



No modo image-to-image, foram apresentadas diversas técnicas úteis para refinar ou estilizar imagens, permitindo transformar uma imagem de referência em uma nova versão visual, mantendo a coerência estrutural. Durante o processo, o parâmetro *denoise strength* foi demonstrado na prática, evidenciando que, nesse modo, exerce grande impacto no controle do nível de transformação e na fidelidade ao estilo original. Essa etapa do manual reforça o papel da IA gerativa como ferramenta criativa, na qual ajustes graduais produzem resultados mais consistentes e personalizados.

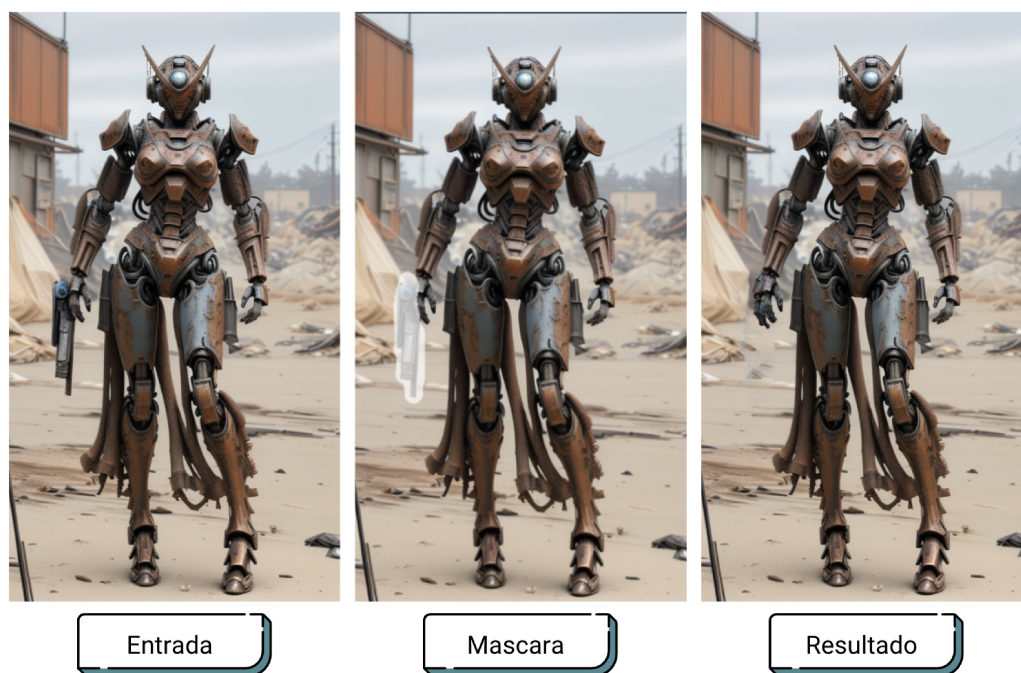
A Figura 12 apresenta a interface do ComfyUI com o *workflow* de image-to-image. Como imagem de entrada, utilizou-se a obra de Franklin Cascaes, "Maricota", transformada em uma versão robótica, facilitando a geração do modelo 3D subsequente. As obras de Franklin Cascaes, renomado artista de Santa Catarina, são reconhecidas por registrar com riqueza de detalhes a cultura, o folclore e as tradições locais, incluindo personagens, lendas e cenas do cotidiano. Elas representam um importante legado cultural, constituindo uma referência relevante para pesquisas e criações artísticas baseadas na tradição e folclore de Santa Catarina. Para o exemplo prático do manual, escolheu-se a "Maricota", devido às suas características humanoides. Essa forma facilita a aplicação de esqueleto durante a modelagem 3D, tornando o processo de animação mais simples e eficiente.

Figura 12 – Exemplo de geração de imagem - ComfyUI.



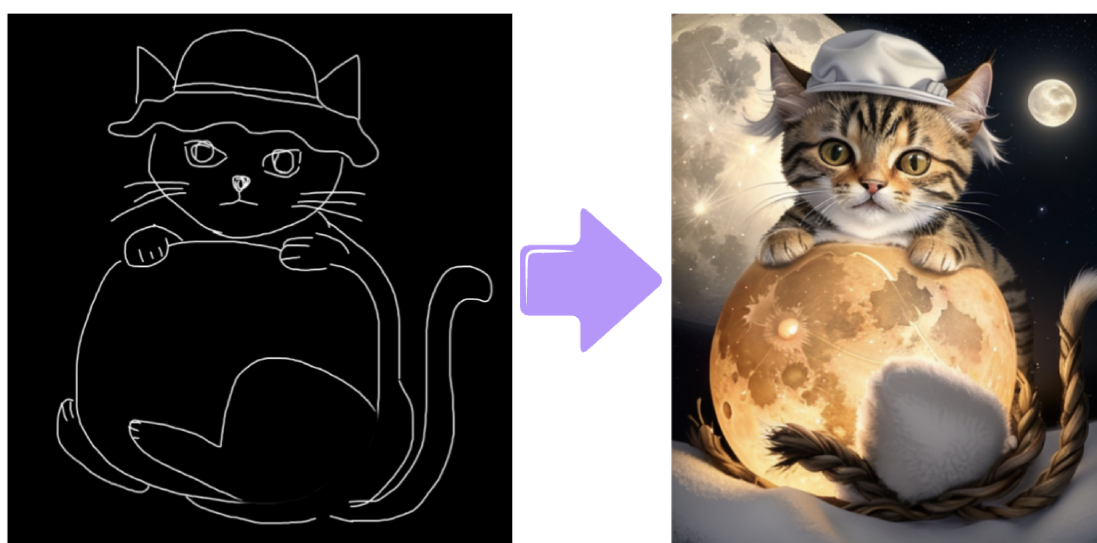
O inpainting foi empregado de maneira direcionada para corrigir áreas específicas, garantindo a continuidade visual antes da modelagem 3D. No manual, apresenta-se um exemplo prático (Figura 13): a partir de uma imagem gerada via image-to-image, aplicou-se uma máscara sobre a mão da personagem que segurava uma arma, junto com um prompt solicitando a remoção do objeto. O resultado foi uma nova imagem da mesma personagem, agora sem a arma, preservando sua integridade visual.

Figura 13 – Exemplo prático de inpainting do manual.



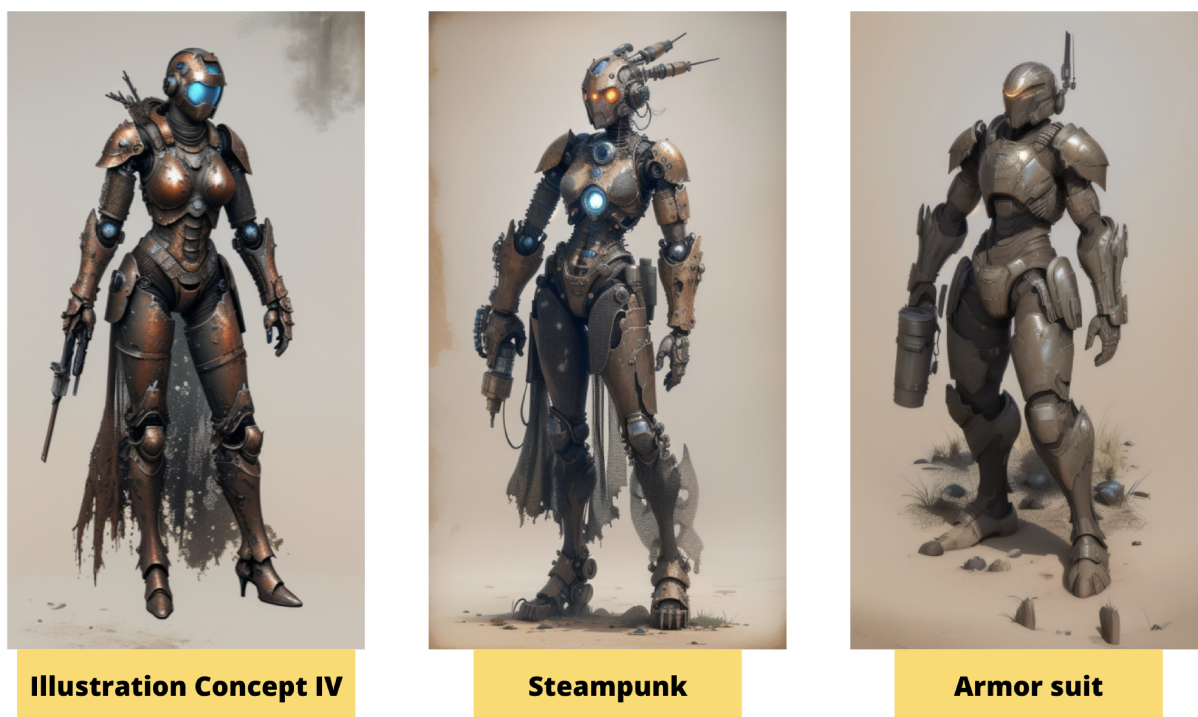
Em seguida, foram apresentados exemplos práticos utilizando o ControlNet, com os modelos OpenPose, Canny Edge, Scribble e Depth Map. Cada modelo possui características e objetivos distintos, permitindo que o usuário selecione a opção mais adequada ao tipo de controle desejado. A Figura 14 ilustra um exemplo prático com o modelo Scribble: a imagem de entrada foi um desenho feito manualmente, e o resultado manteve fielmente as características estruturais do esboço original.

Figura 14 – Exemplo prático de ControlNet Scribble do manual.



Por fim, o manual aborda a integração de modelos LoRA, módulos leves que permitem aplicar estilos, temáticas e identidades visuais específicas aos modelos base. São apresentados sites que disponibilizam diversas opções de modelos LoRA treinados para download, além de instruções sobre como inseri-los e utilizar a palavra de ativação correspondente no prompt, ajustando o peso de influência para calibrar a intensidade do estilo sobre a imagem final. Para exemplificar, o manual mostra resultados obtidos ao aplicar diferentes modelos LoRA à mesma imagem de entrada, a personagem "Maricota", evidenciando como a técnica permite variações estilísticas mantendo a estrutura original da referência (Figura 15).

Figura 15 – Exemplos de resultados de LoRA do manual.



Assim, a seção de geração de imagens do manual prático oferece um guia completo para a criação de ativos visuais de alta qualidade, detalhando desde a preparação das imagens e o ajuste de parâmetros até o uso de técnicas como Image-to-Image, Inpainting, ControlNet e LoRA. Cada técnica é apresentada com exemplos ilustrativos e instruções passo a passo, disponíveis tanto nas interfaces Automatic1111 quanto ComfyUI. Dessa forma, o usuário pode escolher a ferramenta de sua preferência e compreender como aplicar cada recurso para refinar, estilizar e estruturar imagens de referência, garantindo consistência e qualidade visual antes da posterior modelagem 3D.

## 4.2 GERAÇÃO E AJUSTE DE MODELOS 3D

Após a geração e o refinamento das imagens de referência, a etapa seguinte do manual prático aborda a criação e o ajuste de modelos 3D a partir dessas imagens. Aqui são detalhadas as técnicas e ferramentas utilizadas para transformar uma imagem de personagem em modelos 3D consistentes. São apresentadas tanto a criação inicial dos modelos por meio de ferramentas de IA quanto as técnicas de refinamento e correção necessárias para garantir qualidade visual e consistência estrutural.

### 4.2.1 Hunyuan 3D 2.0

O primeiro modelo abordado para a geração de modelos 3D é o Hunyuan 3D 2.0, desenvolvido pela Tencent (Team, 2025). Esta ferramenta permite criar malhas tridimensionais otimizadas e gerar mapas de textura realistas a partir de uma única imagem 2D. De acordo com o artigo publicado pela *Towards Deep Learning*, o Hunyuan 3D 2.0 representa um avanço significativo ao democratizar uma tecnologia antes acessível apenas a profissionais especializados, além de abrir inúmeras possibilidades para áreas como impressão 3D, jogos, cinema e animação, contribuindo para a redução de custos na produção de ativos tridimensionais.

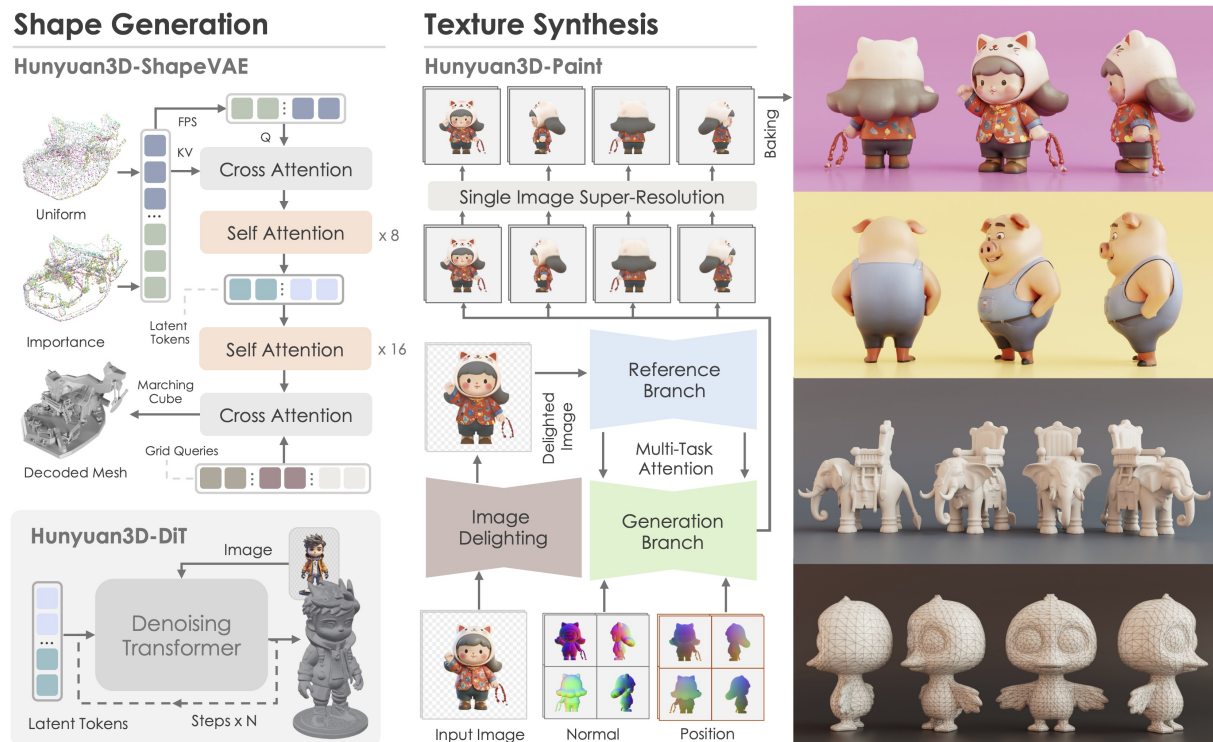
O Hunyuan 3D 2.0 apresenta um *pipeline* de geração em duas etapas, iniciando pela criação de uma malha base sem textura (Hunyuan 3D-ShapeVAE e Hunyuan 3D-DiT), seguida pela síntese do mapa de textura (Hunyuan 3D-Paint) correspondente a essa malha (Team, 2025). Assim, a arquitetura pode ser detalhada em três técnicas principais:

1. **Hunyuan 3D-ShapeVAE:** Nesta etapa inicial, utiliza-se uma representação compacta dos objetos 3D, transformando formas existentes em tokens. Esse processo é realizado com amostragem por importância, dando maior atenção a bordas, cantos e áreas detalhadas, garantindo que os elementos finos do modelo sejam preservados (Chokhra, 2025). Ao converter as formas tridimensionais em uma representação processável pela IA, o sistema permite resultados mais precisos e detalhados.
2. **Hunyuan 3D-DiT:** Enquanto o Hunyuan 3D-ShapeVAE gera os tokens correspondentes, outro fluxo analisa a imagem de entrada. Os dois fluxos comunicam-se por mecanismos de atenção, permitindo que a geração da malha considere detalhes e relações espaciais de forma integrada. Para garantir suavidade e fidelidade na reconstrução, aplica-se a técnica de *flow-matching*, um método avançado de difusão (Chokhra, 2025). Como resultado, o sistema produz uma malha tridimensional de alta qualidade em formato de *wireframe*. A principal inovação do Hunyuan 3D 2.0 é sua capacidade de prever, de maneira inteligente, os lados ocultos do objeto, aumentando a consistência e o realismo do modelo gerado.

3. **Hunyuan 3D-Paint:** Nesta etapa, a malha gerada e a imagem de referência são usadas para criar o modelo texturizado final. O processo envolve limpeza da imagem, escolha de múltiplos ângulos de câmera, geração de vistas múltiplas usando o fluxo duplo da IA e aplicação dessas fotos sobre a malha (*texture baking*). O modelo combina arquitetura de dupla transmissão, atenção multi-tarefa e processamento inteligente, garantindo consistência, alinhamento entre vistas e precisão geométrica (Chokhra, 2025). O resultado é um modelo tridimensional texturizado de alta qualidade, pronto para uso em jogos, animações ou qualquer software 3D.

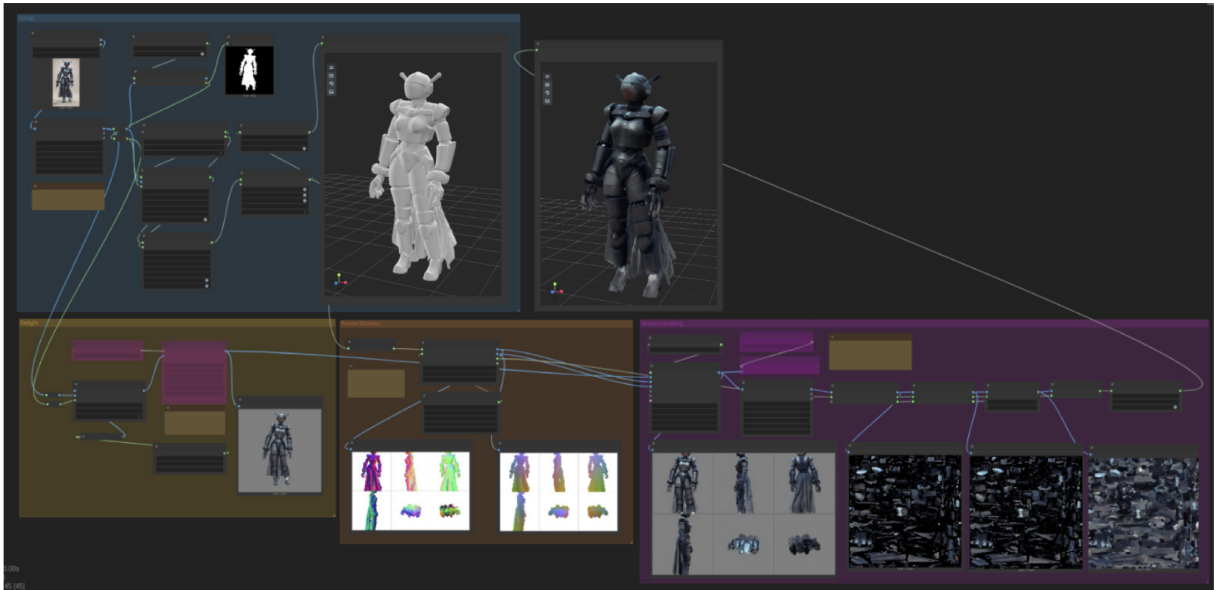
A Figura 16 ilustra essas etapas e a arquitetura do modelo, detalhando a comunicação entre os módulos e a geração final do modelo 3D texturizado.

Figura 16 – Pipeline e arquitetura do Hunyuan 3D 2.0.



Posteriormente, o manual orienta sobre a instalação do modelo e a execução do *workflow* no ComfyUI, incluindo a visualização da malha com textura gerada a partir da imagem produzida na etapa anterior, como ilustrado na Figura 17. Essa representação permite acompanhar de forma clara todas as etapas do processo, desde a análise da imagem de entrada e a geração da malha base pelos módulos ShapeVAE e DiT, até a síntese do mapa de textura final pelo módulo Paint. Além disso, facilita a compreensão da comunicação entre os módulos e do fluxo de dados, evidenciando como cada etapa contribui para a criação de um modelo 3D texturizado de alta qualidade.

Figura 17 – Exemplo de Hunyuan 3D 2.0 no ComfyUI

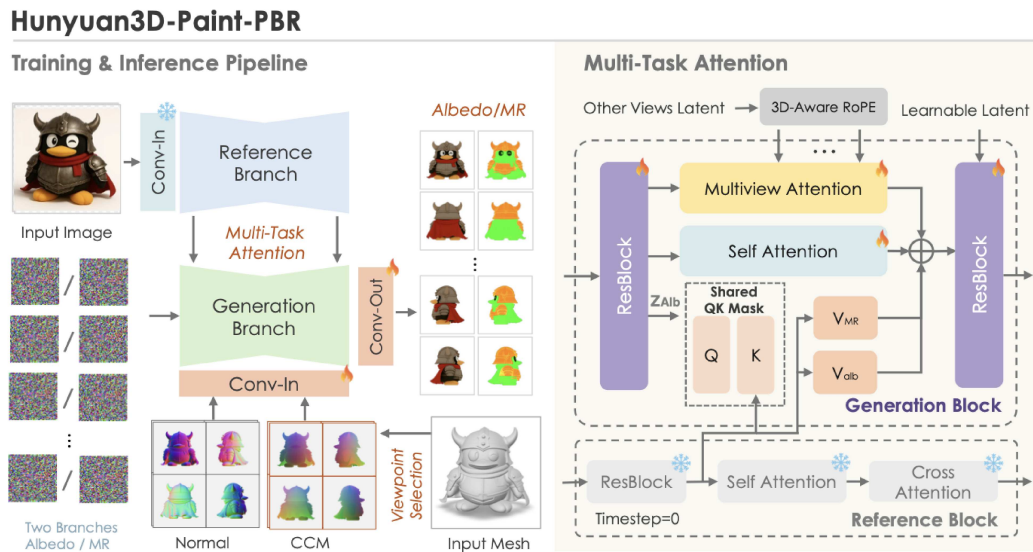


#### 4.2.2 Hunyuan 3D 2.5

Em seguida, o manual apresenta o Hunyuan 3D 2.5, a versão mais recente do modelo desenvolvido pela Tencent, que segue a mesma arquitetura geral do Hunyuan 3D 2.0. Esta versão introduz um novo modelo de geração de formas *lattice*, que é um modelo de difusão em larga escala capaz de produzir formas detalhadas e de alta fidelidade, com bordas nítidas e superfícies suaves, a partir de uma única imagem ou de quatro imagens em múltiplas vistas (Hunyuan3D, 2025). Treinado em um conjunto de dados 3D extenso e de alta qualidade, contendo objetos complexos, o modelo foi projetado para gerar detalhes excepcionais. Para garantir eficiência, são empregadas técnicas de orientação (*guidance*) e destilação de passos (*step distillation*), que reduzem o tempo de inferência (Hunyuan3D, 2025).

Além disso, a geração de texturas do Hunyuan 3D 2.5 é baseada em um *framework* que utiliza mapas normais e *Color Correction Map* gerados pela malha 3D como condições geométricas, juntamente com uma imagem de referência como guia. O modelo produz mapas de materiais PBR de alta qualidade, com consistência entre diferentes vistas, garantindo texturas contínuas e realistas (Hunyuan3D, 2025), como mostrado na Figura 18.

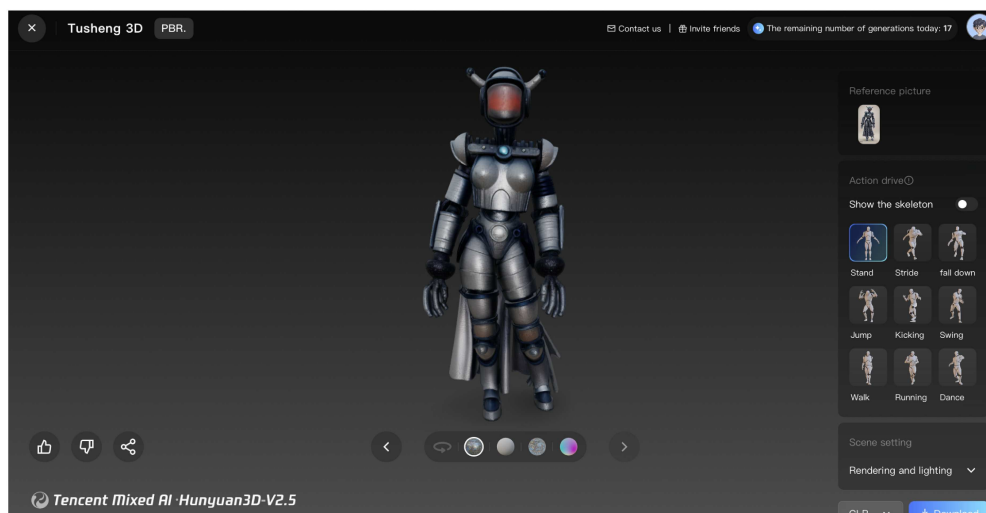
Figura 18 – Visão geral de geração de materiais no Hunyuan 3D 2.5



Em relação ao uso prático do Hunyuan 3D 2.5, atualmente a plataforma disponibiliza um limite diário de 20 gerações por usuário. A interface online permite executar funcionalidades de Image-to-3D, Text-to-3D, *rigging* automático, o conjunto de aplicações 3D (*3D Application Suite*) e a criação de modelos de mundo 3D (*3D World Model*), oferecendo aos usuários formas de testar e explorar os recursos do modelo dentro dessas restrições.

Para ilustrar o funcionamento do Image-to-3D, a Figura 19 apresenta uma malha gerada a partir de uma imagem obtida na etapa anterior, utilizando o site oficial do Hunyuan 3D 2.5. É possível observar a geometria detalhada da malha, que preserva as características estruturais do objeto original.

Figura 19 – Exemplo de uso de Hunyuan 3D 2.5



### 4.2.3 Meshy AI

O manual também apresenta o Meshy AI, um gerador de modelos 3D com inteligência artificial que auxilia artistas 3D, desenvolvedores de jogos, entusiastas de impressão 3D e criadores de protótipos em *Extended Reality* (XR) a transformar textos e imagens em modelos 3D de alta qualidade. Conforme documentado no site oficial, o Meshy AI oferece recursos como Text-to-3D, Image-to-3D, Text-to-Texture e opções de animação, tudo isso com interface amigável para usuários com pouca experiência (AI, 2025).

A plataforma é acessível diretamente pelo navegador e não requer instalação local, o que facilita seu uso. O plano gratuito oferece 200 créditos e 10 downloads por mês, sendo que cada geração consome 10 créditos.

O manual prático também contém instruções de uso do Meshy AI, permitindo que o usuário experimente a geração de modelos 3D e compare seus resultados com aqueles obtidos via Hunyuan 3D, oferecendo mais opções e flexibilidade na criação de ativos tridimensionais. A Figura 20 apresenta o resultado da malha 3D com textura gerada a partir do Meshy AI.

Figura 20 – Exemplo de uso de Meshy AI



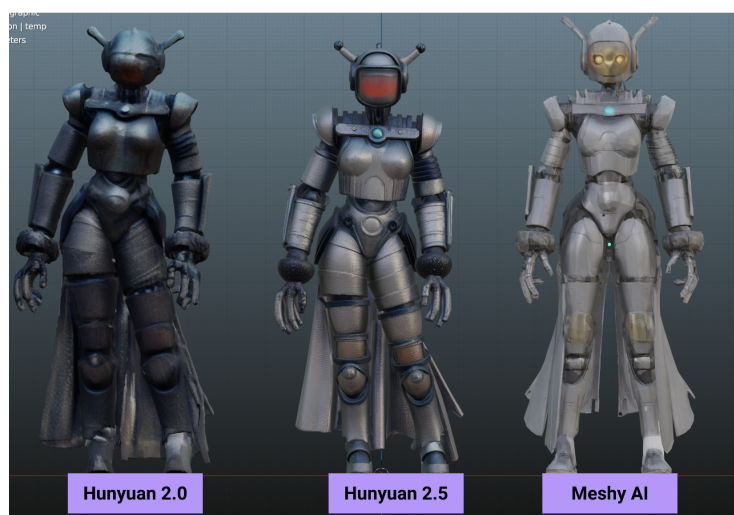
Essas opções permitem explorar diferentes fluxos de trabalho e resultados sem a necessidade de hardware avançado. Vale destacar que o Hunyuan 3D 2.0, quando executado localmente pelo ComfyUI, requer um computador equipado com placa de vídeo (GPU) com, no mínimo, 8GB de VRAM, além da configuração adequada do ambiente, o que pode restringir seu uso a usuários com maior capacidade técnica e recursos de hardware. Dessa forma, a inclusão de plataformas online serve para oferecer flexibilidade e acessibilidade na criação de modelos 3D.

A Figura 21 apresenta três personagens 3D geradas a partir da mesma imagem de entrada, obtida no processo de image-to-image com a obra "Maricota". As comparações

entre os modelos evidenciam diferenças na qualidade da malha e das texturas:

- **Hunyuan 3D 2.0:** apresenta pequenas falhas na textura, e as mãos da personagem aparecem coladas ao corpo.
- **Hunyuan 3D 2.5:** gera malha com qualidade superior, com texturas mais detalhadas e mãos separadas, mantendo maior fidelidade à imagem de referência.
- **Meshy AI:** produz uma malha de boa qualidade, porém a textura não reproduz fielmente a imagem de entrada.

Figura 21 – Comparação do resultado entre 3 modelos



Essa comparação evidencia que cada ferramenta possui características distintas quanto à fidelidade das texturas, à qualidade da malha e ao nível de detalhamento, oferecendo ao usuário diferentes alternativas para a geração de modelos 3D.

#### 4.2.4 Correção e Ajuste de Modelos 3D

Após a geração dos modelos 3D pela IA gerativa, o manual prático também aborda a necessidade de ajustes para corrigir imperfeições que podem ocorrer durante o processo, como, por exemplo, personagens gerados pelo Hunyuan 3D 2.0 com as mãos coladas ao corpo ou falhas nas texturas. Ou seja, os modelos podem apresentar problemas como geometria irregular, malhas interpenetradas ou texturas mal aplicadas. Para esses casos, o manual apresenta um guia detalhado de uso do Blender, permitindo que o usuário realize as correções necessárias e prepare os modelos para a aplicação do esqueleto de animação.

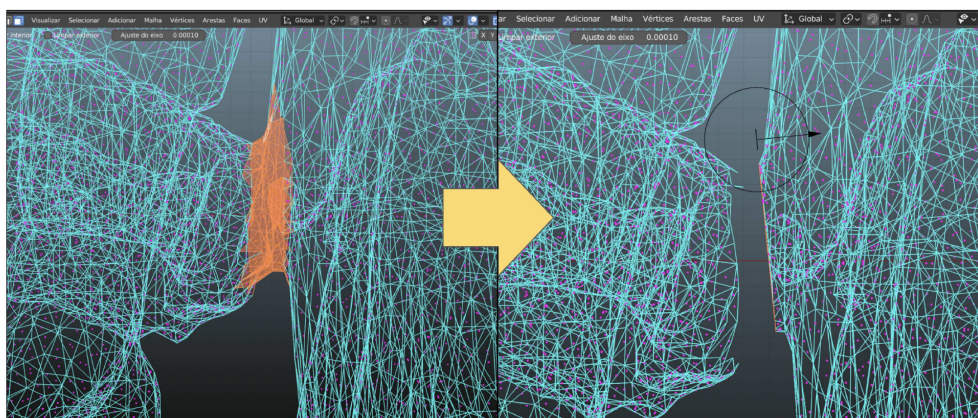
Um dos problemas mais comuns é a malha colada, que pode ocorrer dependendo da IA gerativa utilizada e da imagem de entrada. Por exemplo, se a imagem do personagem apresenta partes do corpo próximas, como as mãos próximas ao tronco, a geração da

geometria pode resultar em malhas coladas. O manual apresenta duas abordagens para corrigir esse problema:

- **Deletar faces diretamente:** selecionar as faces a serem removidas e, em seguida, fechar os buracos criados, gerando novas faces para unir corretamente as partes separadas.
- **Deletar usando Knife Bisset:** ferramenta do Blender que permite realizar cortes retos e precisos, ideal para regiões mais complexas.

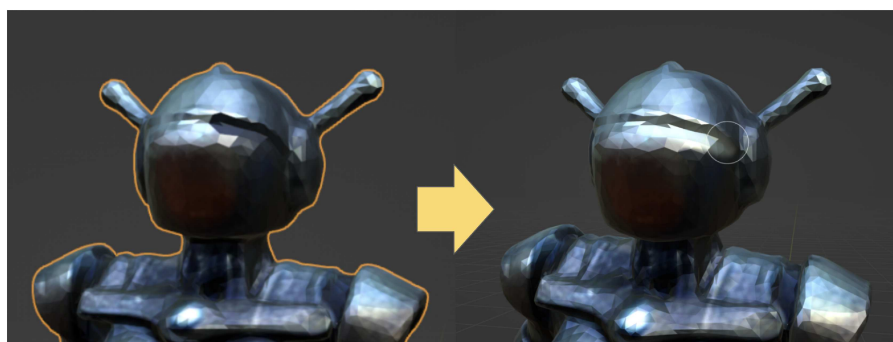
A Figura 22 apresenta um exemplo de malha colada gerada pela IA, mostrando como as partes do modelo podem ficar unidas incorretamente (à esquerda) e o resultado após a correção com as técnicas ensinadas no manual (à direita).

Figura 22 – Exemplo de malha colada e malha corrigida no Blender



Outro problema comum são texturas incorretas ou incompletas. Para ajustá-las, o manual recomenda o Modo de Pintura de Textura, utilizando a ferramenta Besuntar (*Smear*) para espalhar cores de forma natural e corrigir imperfeições. A Figura 23 ilustra a textura da cabeça do personagem antes e após a correção. Após os ajustes, o UV Map deve ser salvo para preservar as modificações realizadas.

Figura 23 – Exemplo de textura incompleta e corrigida no Blender



Por fim, pode ser necessário ajustar o *Shader*, que é o conjunto de propriedades que define como a superfície do modelo interage com a luz, determinando aparência como brilho, transparência, metalicidade e rugosidade. O manual detalha como configurar os parâmetros de material, comparando o visual do personagem antes e depois do ajuste (Figura 24), garantindo um resultado final mais realista e coerente com a textura aplicada.

Figura 24 – Exemplo de ajuste de Shader no Blender



### 4.3 APLICAÇÃO DE ESQUELETOS E ANIMAÇÃO

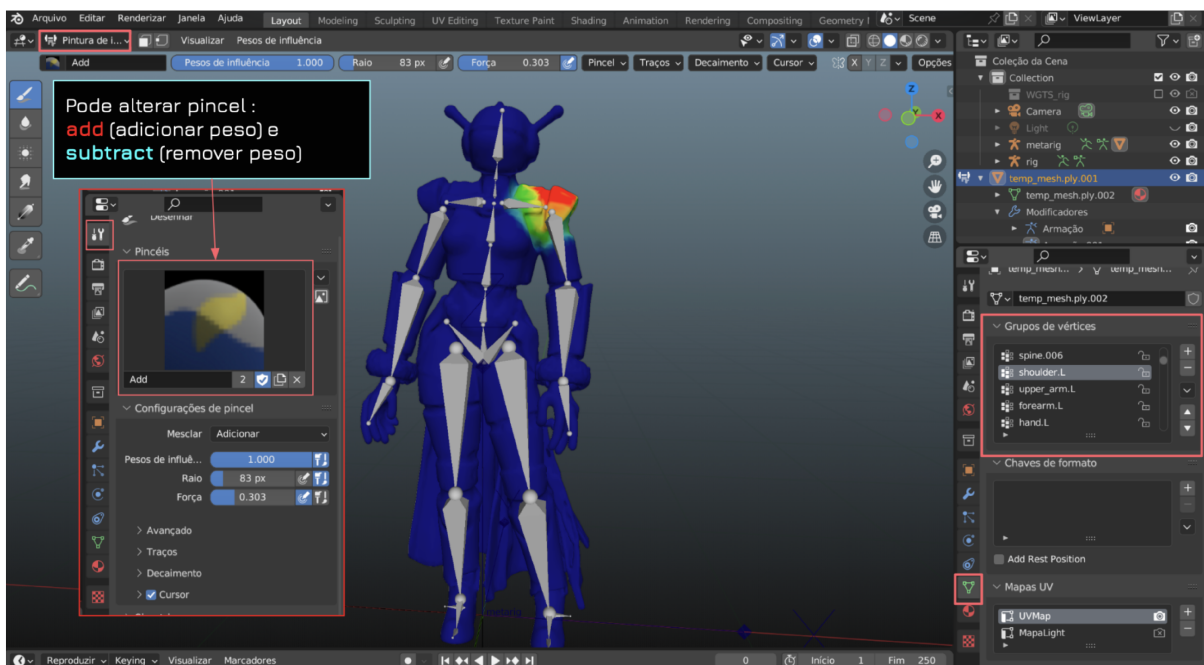
Após a geração e correção dos modelos 3D, o próximo passo detalhado no manual é a aplicação de esqueleto, chamado de *rigging* e animação, que permite que os personagens realizem movimentos andar, correr, pular ou realizar acrobacias em ambiente virtual. Esta etapa pode ser realizada de forma manual, utilizando ferramentas como Blender, ou de maneira automática, por meio de plataformas online.

No *rigging* manual, o esqueleto é ajustado para coincidir com a malha do personagem, garantindo que ossos e vértices estejam corretamente alinhados. O processo envolve:

- **Ajuste do Esqueleto:** alinhar corretamente à malha do personagem antes da vinculação. O objetivo é assegurar que a estrutura óssea corresponda à anatomia do modelo, evitando deformações incorretas durante a animação.
- **Conexão com a malha:** associação da malha ao esqueleto, utilizando pesos de influência automáticos.

- **Correção de influência de pesos:** quando a influência automática dos ossos sobre a malha não é aplicada de forma precisa, é necessário realizar ajustes manuais usando a ferramenta de pintura de pesos (*Weight Painting*), como apresentado na Figura 25. Essa etapa corrige deformações incorretas durante a movimentação do personagem, garantindo que cada parte da malha responda adequadamente ao osso correspondente.
- **Exportação:** salvamento do modelo *rigged* no formato .fbx para uso na Unreal Engine.

Figura 25 – Blender - pintura de pesos



Na aplicação automática, ferramentas como Mixamo permitem a criação rápida de esqueletos para personagens humanoides, oferecendo também animações pré-definidas. O Blender, por sua vez, disponibiliza recursos de auto-rigging através do add-on nativo Rigify, que gera rigs automaticamente a partir de um metarig posicionado sobre o modelo, possibilitando ajustes finos e maior controle sobre o esqueleto e a deformação da malha. Outras plataformas de IA, como Hunyuan 3D 2.5 e Meshy AI, permitem gerar modelos já com esqueleto e animação aplicados. Essa abordagem é mais prática e acessível, mas, por ser automática, pode não se adaptar perfeitamente a todos os modelos, especialmente em poses complexas ou em detalhes como mãos próximas ao corpo, podendo exigir correções manuais.

## 4.4 CRIAÇÃO DE JOGO

A última etapa do manual aborda a criação de jogos 3D, um processo que integra modelagem, animação e programação. Para isso, utiliza-se a Unreal Engine 5, uma das ferramentas mais populares e poderosas da indústria para desenvolvimento de jogos tridimensionais, demonstrando como integrar modelos gerados por IA e aplicar a lógica de jogo interativa. O ambiente utilizado neste trabalho foi a Fortaleza de Anhatomirim, criado no Unreal Engine pelo professor de animação Flávio Andaló, que serviu como base para a construção do protótipo. Foram realizadas modificações, incluindo a alteração de texturas com ativos gerados por IA e adição de novas estruturas, adaptando o cenário aos objetivos do manual e do jogo desenvolvido.

O processo inicia com a importação dos ativos, como malhas, esqueletos e animações em formato .fbx, e a configuração do personagem no *Blueprint* principal do jogo. Em seguida, a animação é controlada dinamicamente por meio do *Animation Blueprint* e das *State Machines*, que definem transições entre estados como parado, correndo ou pulando, com base na velocidade e nas condições do personagem.

Após a inserção de personagens e animações no ambiente, é necessário definir a lógica do jogo, que consiste no conjunto de regras, objetivos e interações que transformam os elementos visuais em um sistema jogável. No manual, a lógica apresentada é propositalmente simples, com o objetivo de demonstrar funcionalidades básicas da Unreal Engine 5, utilizando pontuação e desbloqueio de objetivos como exemplo:

- **Objetivo Final:** O jogador deve alcançar 100 pontos para sair da ilha e vencer o jogo.
- **Sistema de Pontos:**
  - Gema 1 → 1 ponto
  - Gema 2 → 2 pontos
  - Gema 3 → 100 pontos
- **Restrições e Progressão:**
  - A Gema 3 está escondida atrás de uma grade de madeira.
  - Para abrir a grade, o jogador precisa ativar duas plataformas.
  - Cada plataforma só pode ser ativada quando o jogador acumular pelo menos 20 pontos.

Essa estrutura demonstra como regras simples podem ser implementadas para controlar a progressão do jogador e as interações no ambiente de jogo, integrando ativos 3D gerados por IA, animações e colisores de maneira funcional.

Para a criação de ativos, como gemas, plataformas e grades de madeira, foi utilizado o Hunyuan 3D 2.5 no modo text-to-3D, permitindo gerar elementos visuais a partir de descrições textuais. As Figuras 26, 27 e 28 apresentam os ativos gerados, evidenciando a aplicabilidade da IA na produção de elementos visuais para o ambiente do jogo.

Figura 26 – Hunyuan 3D 2.5 - gemas

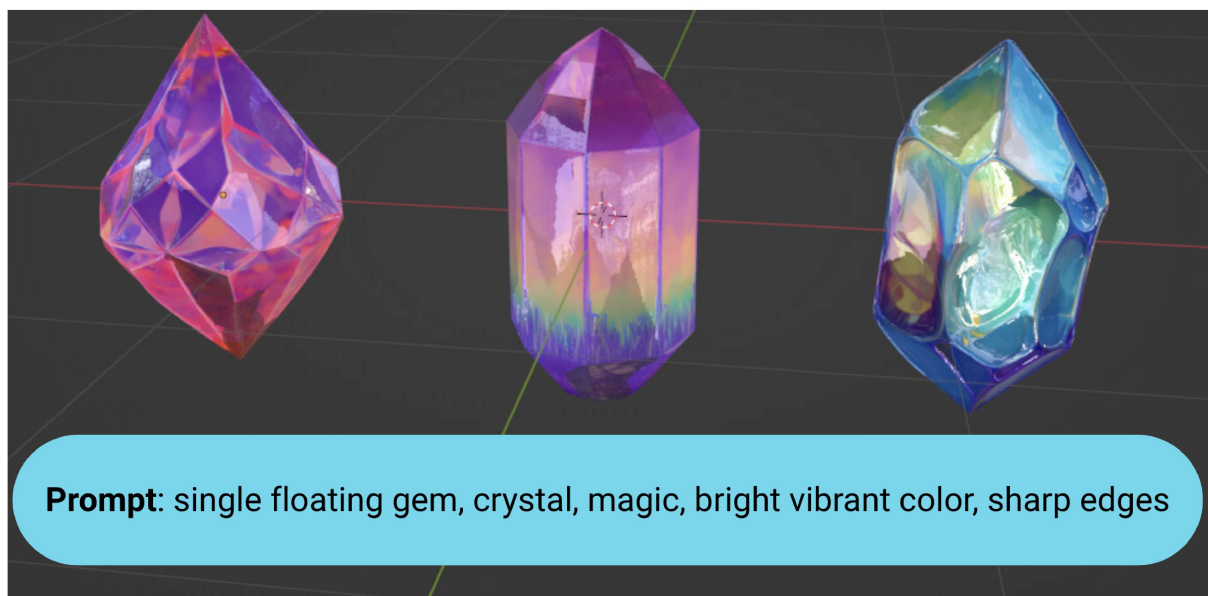


Figura 27 – Hunyuan 3D 2.5 - plataformas

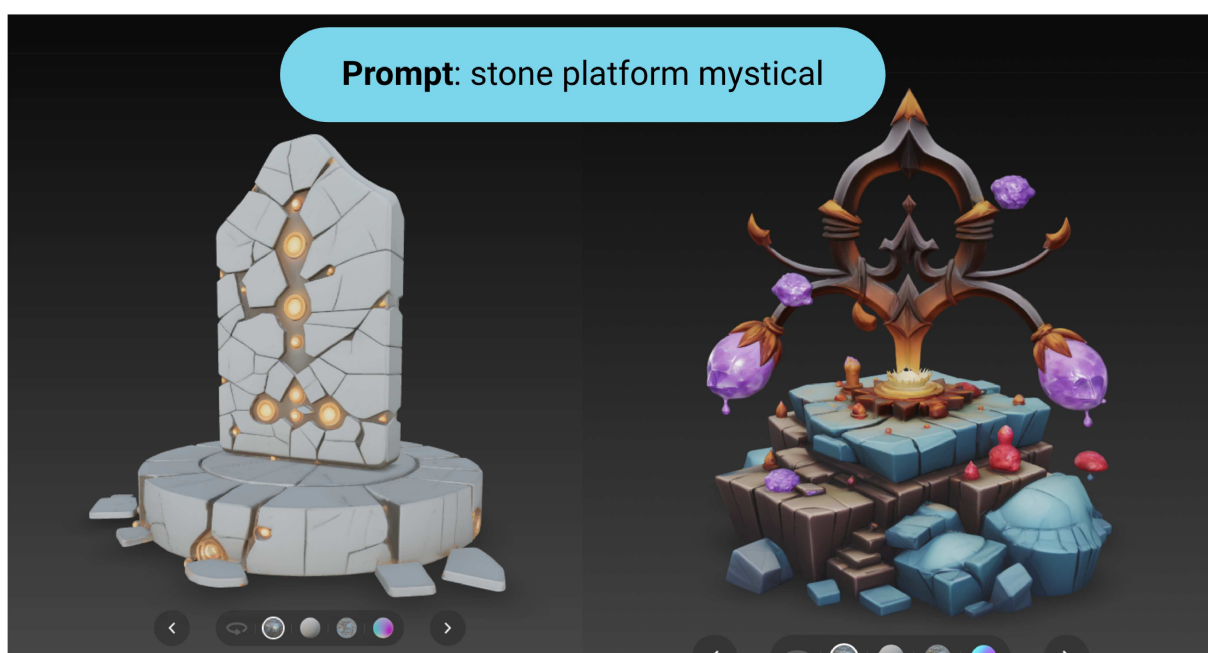


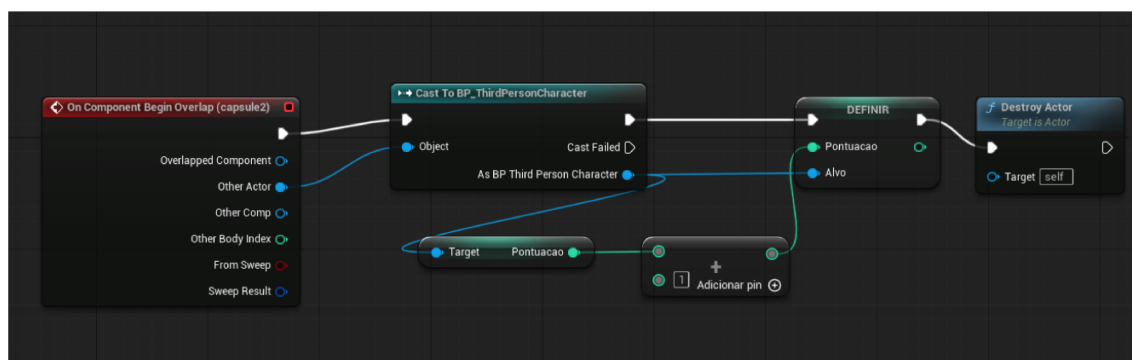
Figura 28 – Hunyuan 3D 2.5 - grade de madeira



A etapa seguinte trata da integração do ambiente e colisão, configurando objetos interativos, como gemas coletáveis, com colidores e eventos de sobreposição que detectam o jogador e atualizam a pontuação. Nessa etapa, foi criada uma classe *Blueprint* do tipo *Actor* para representar a gema coletável no cenário. O *Blueprint* recebeu uma malha estática (a representação visual da gema) e um componente de colisão do tipo *Capsule Collision*, escolhido por se ajustar melhor ao formato do modelo 3D. A malha teve sua predefinição de colisão alterada de “*Block All Dynamic*” para “*No Collision*”, permitindo que o personagem atravessasse o objeto. Já a cápsula de colisão permaneceu ativa para detectar a sobreposição (*overlap*) com o personagem.

A lógica de coleta foi implementada no *Event Graph* do *Blueprint*. Nesse gráfico, o evento de sobreposição foi configurado para verificar se o ator que entrou em contato é o personagem jogável. Quando isso ocorre, o sistema executa duas ações principais: incrementa a pontuação do jogador de acordo com o tipo da gema coletada e, em seguida, remove a gema da cena utilizando o nó *Destroy Actor*. A Figura 29 apresenta o *Event Graph* configurado para o *Blueprint* da gema, destacando os nós responsáveis pela detecção da colisão, atualização da pontuação e destruição do ator.

Figura 29 – Unreal Engine 5 - Blueprint de gemas



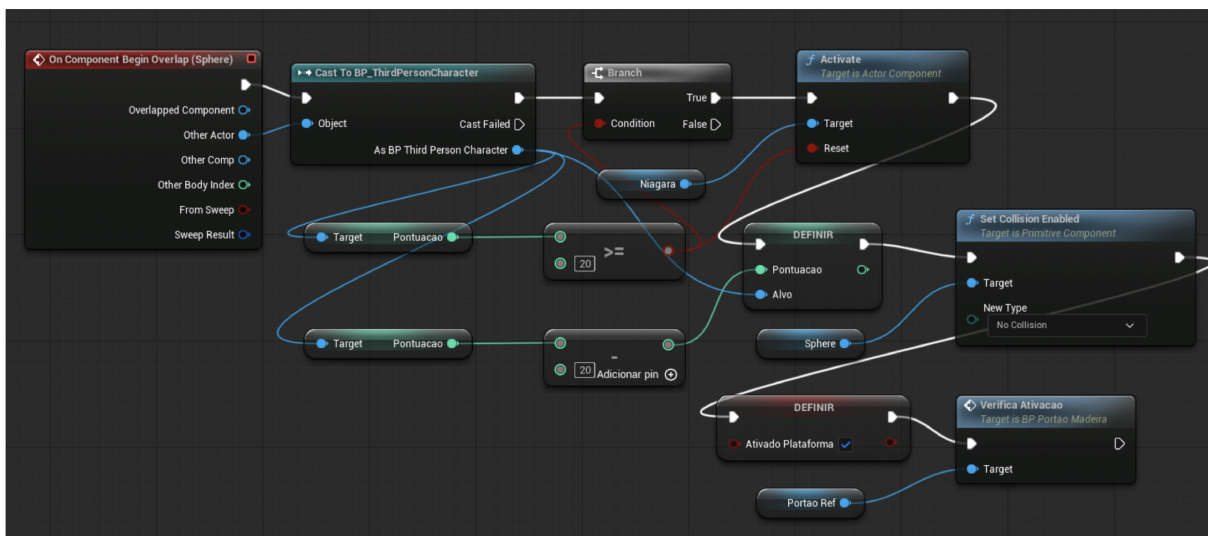
Após a implementação do sistema de coleta e pontuação, foi criada uma interface para exibir, em tempo real, a pontuação atual do jogador. O manual apresenta o uso de um *Blueprint* do tipo *Widget*, responsável por definir os elementos visuais da interface gráfica. Dentro do *Widget Blueprint*, é adicionado um componente de texto que será responsável por exibir o valor da pontuação. Esse componente é configurado com uma vinculação (*Bind*) à variável de pontuação armazenada no *Blueprint* do jogador. Assim, a cada atualização dessa variável, por exemplo, quando o jogador coleta uma gema, o texto do *Widget* é automaticamente atualizado, exibindo o valor correto na tela. Por fim, o *Widget* é adicionado à tela do jogador por meio do nó *Add to Viewport*, dentro do *Event Graph* do *Blueprint* do jogador, garantindo que a interface seja renderizada continuamente durante a execução do jogo.

A próxima etapa aborda a interação entre o jogador e os elementos do ambiente, como as plataformas de ativação e a grade de madeira que bloqueia o acesso à gema final. Cada plataforma pode ser ativada somente quando o jogador acumula pelo menos 20 pontos. No *Event Graph* da plataforma, um nó *Branch* realiza a verificação da pontuação atual. Caso a condição seja satisfeita, o sistema executa uma sequência de ações:

1. Ativa um efeito visual utilizando o sistema *Niagara*, simulando energia ou brilho para indicar que a plataforma foi acionada;
2. Decrementa 20 pontos da pontuação total do jogador, representando o custo da ativação;
3. Desabilita a colisão da plataforma, impedindo que o jogador a reative;
4. Define a variável booleana *AtivadoPlataforma* como *true*, indicando que a plataforma correspondente já foi acionada.
5. Chama a função da grade de madeira responsável por verificar se ambas as plataformas foram ativadas, liberando a gema final caso a condição seja satisfeita.

O sistema *Niagara* é utilizado para criar efeitos visuais dinâmicos e responsivos dentro da Unreal Engine 5, permitindo adicionar partículas, luzes e movimentos que reforçam visualmente as ações do jogador. Nesse contexto, sua aplicação aumenta a imersão e fornece um retorno visual imediato quando uma plataforma é ativada. A Figura 30 apresenta o *Event Graph* do *Blueprint* da plataforma.

Figura 30 – Unreal Engine 5 - Blueprint de plataforma



A grade de madeira é configurada para se destruir automaticamente após a ativação das duas plataformas. O *Blueprint* responsável pela grade contém referências diretas aos atores correspondentes às plataformas, permitindo monitorar o estado de cada uma. Uma função personalizada realiza a verificação das variáveis booleanas *AtivadoPlataforma1* e *AtivadoPlataforma2*. Quando ambas são verdadeiras, o sistema executa o comando *Destroy Actor*, removendo a grade de madeira e liberando o caminho para que o jogador alcance a gema de maior valor. A Figura 31 ilustra a função de verificação implementada na grade de madeira, responsável por controlar sua destruição condicional.

Figura 31 – Unreal Engine 5 - Blueprint de grade de madeira (função de verificação)



Dessa forma, os *Blueprints* transformam os ativos visuais gerados por IA em um sistema jogável completo, integrando colisões, efeitos visuais, pontuação, progressão e interface de usuário de forma organizada e visualmente compreensível. Essa abordagem facilita testes, ajustes e manutenção, pois todas as regras e interações podem ser visualizadas

como fluxos conectados de nós, tornando o processo mais intuitivo para desenvolvedores iniciantes ou designers sem experiência em programação.

Os últimos capítulos do manual apresentam os recursos visuais que aprimoram a imersão. O uso da ferramenta *Foliage Tool* permite adicionar vegetação, pedras e outros elementos de cenário de forma ágil. Além da rapidez, a ferramenta possibilita ajustar parâmetros como densidade, escala, rotação e alinhamento ao terreno, garantindo variações que tornam o ambiente mais realista. Essa abordagem é especialmente útil para criar florestas, jardins ou terrenos complexos sem exigir grande esforço manual, permitindo que designers e desenvolvedores foquem na composição estética e na jogabilidade.

Além disso, o manual detalha o processo de criação de *materials*, permitindo configurar a aparência e as propriedades de superfície dos objetos. Por meio do editor de materiais da Unreal Engine, é possível definir cor, rugosidade, reflexão e comportamento metálico, garantindo que elementos como pedras, madeira ou superfícies metálicas respondam à iluminação de forma realista. O manual também apresenta técnicas avançadas para aplicação de texturas realistas, permitindo criar superfícies mais naturais e detalhadas:

- **Tiling (Repetição):** técnica que ajusta quantas vezes uma textura é repetida sobre uma malha 3D. É especialmente útil em superfícies grandes, evitando que a textura pareça esticada ou distorcida. Por exemplo, uma parede de tijolos pode repetir a textura várias vezes para manter o padrão consistente.
- **Texture Bombing:** consiste em projetar a mesma textura diversas vezes, aplicando variações aleatórias de posição, rotação e escala. Essa técnica quebra o padrão de repetição visível, criando um efeito mais orgânico e natural, como se cada elemento da textura fosse ligeiramente diferente do outro. A Figura 32 apresenta uma comparação entre o uso de tiling e texture bombing, evidenciando como cada técnica altera a percepção visual da superfície.
- **Mesclagem de Texturas (Texture Blending):** utiliza o nó Lerp (*Linear Interpolate*), junto a uma máscara em Alpha, para combinar duas texturas diferentes de forma precisa. Isso permite, por exemplo, aplicar musgo sobre um material de tijolo ou areia sobre uma pedra, garantindo que a transição entre texturas seja suave e controlada. A Figura 33 mostra um exemplo prático de mesclagem de texturas, destacando a transição gradual entre materiais distintos.

Figura 32 – Unreal Engine 5 - Comparação entre tiling e bombing

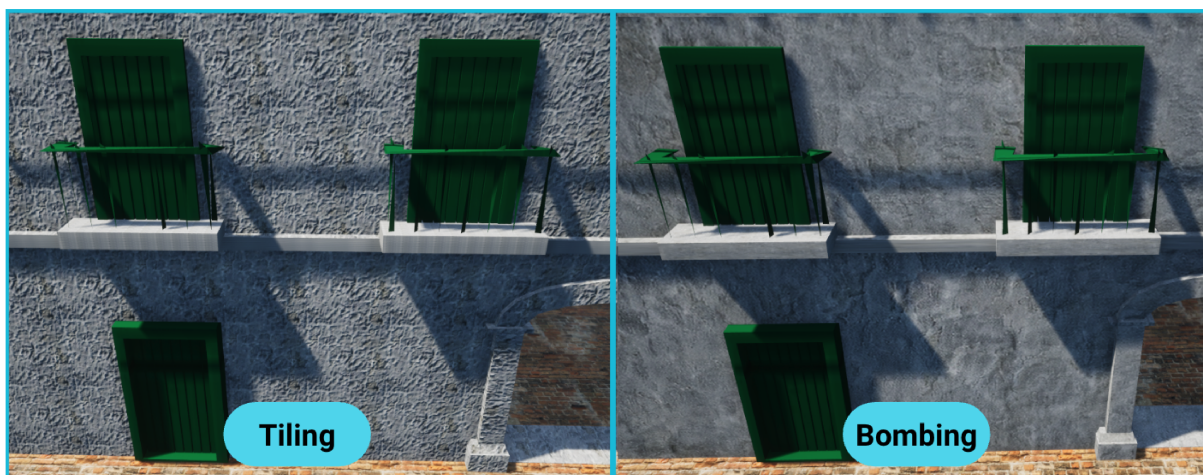
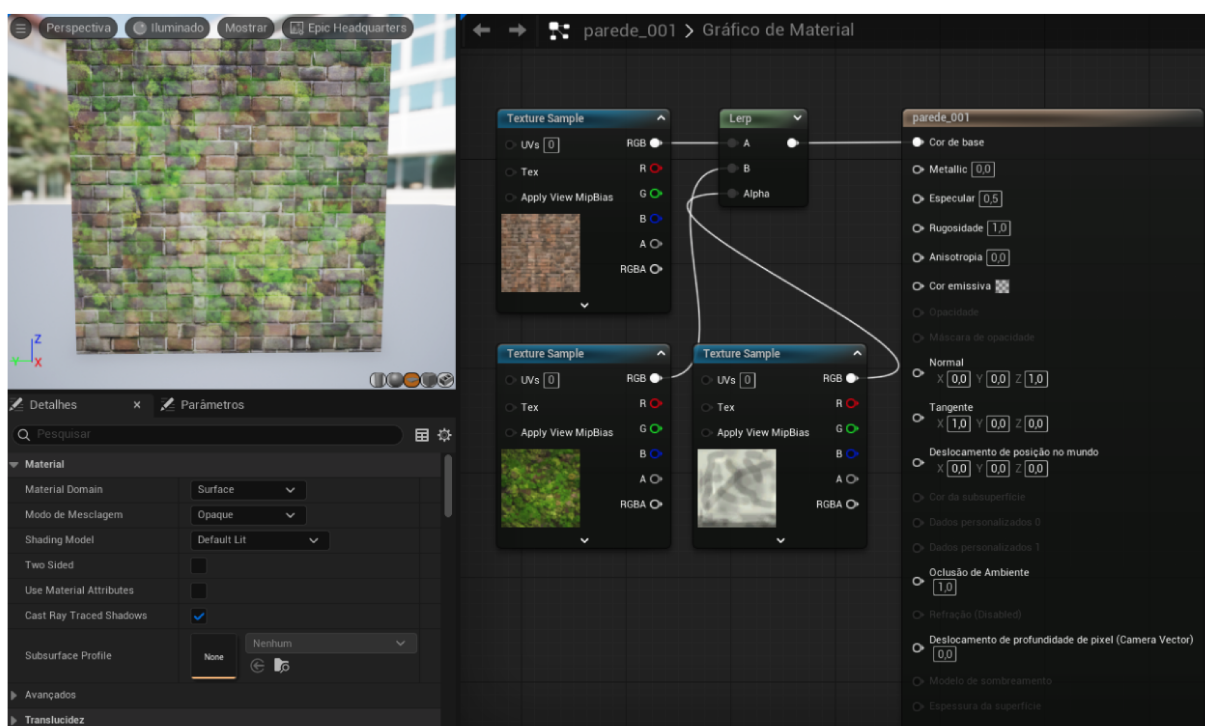


Figura 33 – Unreal Engine 5 - Mesclagem de Textura



Essas técnicas, detalhadas no manual, são fundamentais para aprimorar a qualidade visual dos ambientes e objetos em jogos, tornando-os mais realistas e imersivos. As etapas demonstram como a Unreal Engine 5 integra, de forma eficiente, ativos 3D gerados por IA, animações e lógica de jogo, resultando em um ambiente interativo e visualmente consistente.

Para ilustrar os resultados obtidos, as Figuras 34, 35, 36, 37, 38 e 39 apresentam capturas de tela do jogo desenvolvido a partir do manual proposto, que utilizou como base o projeto do professor Flávio. As imagens exibem o personagem principal, o cenário e a

integração dos elementos criados com o uso de ferramentas de IA gerativa, evidenciando a aplicação prática das técnicas apresentadas. A Figura 36, que representa a casa da comandante, foi inteiramente criada e adicionada ao ambiente, enquanto os demais elementos tiveram apenas suas texturas substituídas por versões geradas por IA gerativa. A única exceção é a igreja (Figura 37), que manteve sua estrutura e textura originais.

O projeto completo do jogo desenvolvido na Unreal Engine 5 está disponível para consulta e download no Apêndice C.

Figura 34 – Personagem inspirada em Maricota



Figura 35 – Ambiente do jogo - Fortaleza de Anhatomirim portada



Figura 36 – Ambiente do jogo - Fortaleza de Anhatomirim casa do comandante



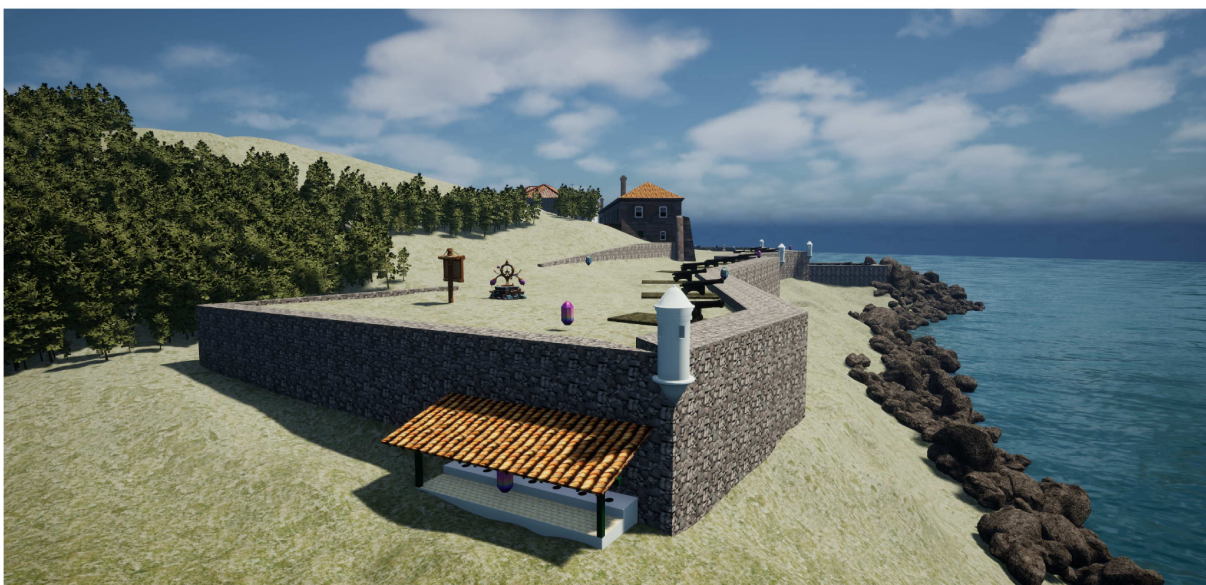
Figura 37 – Ambiente do jogo - Fortaleza de Anhatomirim igreja



Figura 38 – Ambiente do jogo - Fortaleza de Anhatomirim ponta do quartel



Figura 39 – Ambiente do jogo - Fortaleza de Anhatomirim na ponta



## 5 RESULTADOS E LIMITAÇÕES DO USO DE IA GERATIVA

A aplicação de ferramentas de IA gerativa no processo de criação de conteúdos visuais trouxe resultados expressivos em termos de agilidade, variedade e experimentação criativa. Este capítulo apresenta uma análise dos principais pontos fortes e limitações observados durante o desenvolvimento do manual prático e na geração de modelos 3D utilizados no jogo.

### 5.1 PONTOS FORTES OBSERVADOS

Durante a experimentação prática, observou-se que as ferramentas de IA gerativa apresentaram diversos benefícios que contribuíram para o processo criativo na produção de ativos visuais. Entre os principais pontos fortes, destaca-se a agilidade na geração de ideias e conceitos, permitindo ao usuário explorar rapidamente diferentes possibilidades estéticas e estilísticas. Esse ganho de velocidade reduziu significativamente o tempo necessário para criar e testar propostas visuais, especialmente nas fases iniciais de concepção de personagens.

Outro aspecto relevante foi a diversidade e riqueza dos resultados obtidos. O uso de diferentes modelos de difusão, como Realistic Vision, DreamShaper e RevAnimated, proporcionou uma ampla variação de estilos e níveis de realismo, ampliando as possibilidades de experimentação artística. A integração de modelos LoRA permitiu combinar estilos específicos, personalizando ainda mais os resultados. O ControlNet se mostrou uma ferramenta valiosa para aumentar o controle sobre a composição, poses e contornos das imagens, enquanto os recursos de inpainting possibilitaram ajustes precisos e correções localizadas, garantindo maior fidelidade e coerência visual. Essa característica contribuiu para uma abordagem mais exploratória e iterativa do processo criativo, na qual o usuário pode avaliar, comparar e refinar ideias de forma dinâmica.

Além da geração de imagens 2D, o manual também explorou o uso de IA gerativa para criação de modelos 3D. Essa abordagem demonstrou grande potencial na geração rápida de malhas e texturas, permitindo que usuários transformassem conceitos visuais em ativos tridimensionais de maneira eficiente. Entre os principais pontos positivos, destaca-se a agilidade na prototipagem de modelos 3D, que reduz significativamente o tempo necessário em comparação com a modelagem manual tradicional. Além disso, essa ferramenta possibilita que usuários sem conhecimento prévio em modelagem 3D possam gerar personagens, cenários e objetos apenas a partir de ideias ou referências visuais, democratizando o acesso à criação de conteúdos tridimensionais.

## 5.2 LIMITAÇÕES

Apesar dos avanços e benefícios identificados, o uso de IA gerativa também apresentou limitações técnicas e conceituais que precisam ser consideradas para um uso eficaz dessas tecnologias. Uma das principais dificuldades foi o controle limitado sobre os resultados. Mesmo com o uso de prompts detalhados e ajustes de parâmetros, as imagens geradas nem sempre refletiam com precisão a intenção do usuário, apresentando variações imprevisíveis.

A geração de modelos 3D por IA ainda apresenta limitações importantes. Modelos como Hunyuan 3D 2.0 conseguem criar malhas e texturas de forma rápida, mas muitas vezes os resultados exigem ajustes manuais para alcançar nível de qualidade profissional, principalmente em detalhes complexos ou proporções precisas. A geometria gerada pode apresentar irregularidades, falhas de topologia, superfícies incompletas ou artefatos indesejados, tornando necessário o uso de softwares de modelagem tradicionais para refinamento. Esses desafios indicam que, embora a IA seja uma ferramenta poderosa de prototipagem, a intervenção humana continua sendo essencial para garantir precisão e qualidade final.

Portanto, apesar de sua eficiência na fase exploratória, a IA gerativa ainda depende fortemente do julgamento humano. O papel do criador continua essencial para selecionar, ajustar e validar os resultados, assegurando coerência estética, intencionalidade e autenticidade no produto final.

A análise dos resultados evidencia que as ferramentas de IA gerativa possuem grande potencial como aliadas na produção de conteúdos digitais, especialmente por sua capacidade de acelerar processos e estimular a experimentação visual. Contudo, suas limitações reforçam a necessidade de uma abordagem crítica e consciente, em que o usuário compreenda o funcionamento da tecnologia, suas implicações éticas e seus limites criativos.

## 6 QUESTÕES ÉTICAS E CUIDADOS NO USO DE IA GERATIVA

A popularização das ferramentas de IA gerativa proporcionou maior praticidade, facilidade e produtividade para diversas áreas criativas e técnicas. Por exemplo, um estudo com desenvolvedores de jogos indicou que mais de 75% dos participantes afirmaram que essas ferramentas ajudam a concluir tarefas com mais rapidez, aumentam a produtividade e melhoram a qualidade do design (Alharthi, 2025). No entanto, também gerou preocupações relacionadas à ética, autoria, originalidade e uso responsável dessas tecnologias. Este capítulo aborda os principais desafios e cuidados necessários para garantir o uso ético e consciente da IA na produção de conteúdos digitais.

Durante o treinamento, os modelos de IA são alimentados com grandes volumes de dados, que podem incluir obras protegidas por direitos autorais, como ilustrações, fotografias, esculturas digitais ou animações. Caso essas obras sejam utilizadas sem autorização dos autores, pode ocorrer violação de direitos autorais, mesmo que o resultado final gerado pela IA seja diferente da obra original. Os designers temem que o uso de conteúdo gerado por IA possa resultar em plágio não intencional ou disputas jurídicas, tornando essencial que os estúdios de jogos implementem políticas claras sobre como a IA gerativa é utilizada e onde se estabelecem os limites da autoria criativa (Alharthi, 2025).

Um caso particular de preocupação envolve a mimese de estilo, fenômeno pelo qual modelos de IA conseguem reproduzir características estilísticas de artistas presentes no conjunto de treinamento, como cores, texturas, pinceladas e composições. Nesse processo, a IA não copia obras específicas, mas gera novas imagens que parecem ter sido criadas por um determinado artista, mesmo sem seu consentimento. Essa capacidade de imitar estilos levanta questões relevantes sobre autoria, originalidade e reconhecimento na produção artística digital (Sarcevic et al., 2024).

Além das questões relacionadas ao treinamento dos modelos, surge também a discussão sobre a autoria e a propriedade dos conteúdos gerados por IA. Atualmente, não há um consenso claro sobre quem deve possuir esses conteúdos, nem sobre como os direitos devem ser distribuídos entre criadores, desenvolvedores de modelos e usuários finais.

No contexto de desenvolvimento de jogos, isso significa que equipes que utilizam IA gerativa para criar personagens, cenários ou texturas devem estar cientes de que, embora o conteúdo seja original em sua forma final, elementos do treinamento podem ter origem em obras de terceiros. Para garantir o uso ético, legal e responsável de ferramentas de IA gerativa, é fundamental que os usuários conheçam a origem dos modelos e datasets utilizados, privilegiando conteúdos com licenças adequadas ou de domínio público e evitando a utilização de obras protegidas sem autorização. Deve-se manter transparência quanto à participação da IA na criação de ativos, informando quando imagens ou modelos foram gerados com auxílio de IA. A documentação do processo de criação, incluindo *workflows*, parâmetros e imagens de entrada, é essencial para comprovar a originalidade e facilitar a

gestão de possíveis questionamentos legais. Por fim, os usuários devem considerar os impactos sociais e culturais de suas criações, adotando práticas que respeitem a diversidade e mantendo-se atualizados quanto às legislações e diretrizes emergentes sobre o uso de IA gerativa.

## 7 CONCLUSÃO

O desenvolvimento do manual prático permitiu analisar de forma sistemática a aplicação de ferramentas de IA gerativa na criação de conteúdos visuais e modelos 3D para jogos. Através da experimentação com plataformas como Automatic1111 e ComfyUI, foi possível demonstrar como essas tecnologias agilizam a produção, aumentam a diversidade estética e permitem que usuários, mesmo sem experiência prévia em modelagem, transformem ideias em modelos 3D concretos de forma rápida e iterativa. Além disso, a documentação detalhada do processo, incluindo *workflows*, parâmetros de geração e resultados comparativos, contribuiu para a criação de um material educativo que pode servir como referência prática para futuros desenvolvedores e estudantes de computação gráfica e design de jogos.

O manual também evidenciou limitações técnicas, como controle ainda parcial sobre os resultados gerados e a necessidade de ajustes manuais em modelos 3D. Além disso, a experiência prática ressaltou a importância de conhecer a origem dos modelos e datasets, manter transparência quanto à participação da IA e adotar práticas responsáveis e conscientes durante a criação através da ferramenta de IA gerativa. Portanto, reforça-se que a IA gerativa deve ser encarada como uma ferramenta de apoio e não como substituta do processo criativo humano.

Por fim, a integração entre criatividade humana e IA gerativa se mostra capaz de acelerar a produção de conteúdos digitais, democratizar o acesso à criação de elementos visuais e impulsionar a experimentação artística. Entretanto, essa integração requer supervisão constante, reflexão crítica sobre os resultados gerados e responsabilidade ética e legal por parte dos usuários, especialmente considerando aspectos relacionados à autoria, originalidade e licenciamento de conteúdos digitais. O trabalho evidencia que, embora a IA gerativa represente um avanço significativo na criação digital, seu uso pleno e sustentável depende do equilíbrio entre inovação tecnológica e julgamento crítico humano, abrindo caminhos para futuras pesquisas e aplicações em jogos, educação e indústria criativa.

## REFERÊNCIAS

- AHIRWAR, Kailash. **A Very Short Introduction to Diffusion Models**. [S.l.: s.n.], 2023. Disponível em: <https://kailashahirwar.medium.com/a-very-short-introduction-to-diffusion-models-a84235e4e9ae>.
- AI, Meshy. **What is Meshy?** [S.l.: s.n.], 2025. Disponível em: <https://help.meshy.ai/en/articles/9991736-what-is-meshy>.
- ALALAQ, Ahmed Shaker. The History of the Artificial Intelligence Revolution and the Nature of Generative AI Work. **DS Journal of Artificial Intelligence and Robotics**, 2025. DOI: 10.59232/AIR-V2I4P101. Disponível em: [https://www.researchgate.net/publication/388141669\\_The\\_History\\_of\\_the\\_Artificial\\_Intelligence\\_Revolution\\_and\\_the\\_Nature\\_of\\_Generative\\_AI\\_Work](https://www.researchgate.net/publication/388141669_The_History_of_the_Artificial_Intelligence_Revolution_and_the_Nature_of_Generative_AI_Work).
- ALHARTHI, Sultan A. Generative AI in Game Design: Enhancing Creativity or Constraining Innovation?, 2025. DOI: 10.3390/jintelligence13060060. Disponível em: <https://www.mdpi.com/2079-3200/13/6/60>.
- CHOKHRA, Parth. **The Evolution of 3D Generation: Hunyuan-3D 2.0 Explained**. [S.l.: s.n.], 2025. Disponível em: <https://www.towardsdeeplearning.com/the-evolution-of-3d-generation-hunyuan-3d-2-0-explained-74e9ab1e9064>.
- D.GOOGLE. **GAN - Introduction**. [S.l.: s.n.], 2022. Disponível em: <https://developers.google.com/machine-learning/gan>.
- EPSTEIN, Ziv; HERTZMANN, Aaron. Art and the science of generative AI. **Science**, jun. 2023. DOI: 10.1126/science.adh4451. Disponível em: <https://doi.org/10.1126/science.adh4451>.
- HU, Edward et al. LoRA: Low-Rank Adaptation of Large Language Models. **arXiv preprint arXiv:2106.09685**, 2022.
- HUNYUAN3D, Tencent. Hunyuan3D 2.5: Towards High-Fidelity 3D Assets Generation with Ultimate Details, 2025. DOI: 10.48550/arXiv.2506.16504. Disponível em: <https://arxiv.org/pdf/2506.16504>.
- KIBA, Emily Sayuri; WANGENHEIM, Aldo von. **MANUAL DE IA GERATIVA PARA MODELAGEM DE PERSONAGEM 3D**. [S.l.: s.n.], 2025. DOI: 10.13140/RG.2.2.21724.04484. Disponível em: <https://www.incod.ufsc.br/instituto/relatoriostecnicos/relatorios-tecnicos-do-incod-issn-2236-5281-vol-15-2025/>.
- KIBA, Emily Sayuri; WANGENHEIM, Aldo von. Revisão Sistemática do Uso de Inteligência Artificial Gerativa na modelagem 3D para Desenvolvimento de Jogos, 2024. DOI: 10.13140/RG.2.2.32290.03520. Disponível em: <https://www.incod.ufsc.br/wp-content/uploads/2024/06/INCoD-TR-02-2024-P.pdf>.

- KOVÁČIK, Vojtěch. AI-Driven Generation of Defect Product Images, 2024. Disponível em: <https://is.muni.cz/th/14tjv/>.
- LAWLOR, Pat; CHANG, Jerry. **History of AI: How Generative AI Grew from Early Research**. [S.l.: s.n.], 2023. Disponível em: <https://www.qualcomm.com/news/onq/2023/08/history-of-ai-how-generative-ai-grew-from-early-research>.
- LEE, Seongmin et al. Diffusion Explainer: Visual Explanation for Text-to-image Stable Diffusion, 2023. Disponível em: <https://arxiv.org/abs/2305.03509>.
- MISHRA, Onkar. **Stable Diffusion Explained**. [S.l.: s.n.], 2023. Disponível em: <https://medium.com/@onkarmishra/stable-diffusion-explained-1f101284484d>.
- PLOENNIGS, Jonas; BERGER, Matthias. AI art in architecture. **AI in Civil Engineering**, v. 2, n. 8, 2023. DOI: 10.1007/s43503-023-00018-y. Disponível em: <https://doi.org/10.1007/s43503-023-00018-y>.
- ROMBACH, Robin et al. High-Resolution Image Synthesis with Latent Diffusion Models, 2022. Disponível em: <https://arxiv.org/abs/2112.10752>.
- RUDINA SESERI, Kleida Martiro; DOLCE, Kyle. **THE HISTORY OF ARTIFICIAL INTELLIGENCE**. [S.l.: s.n.], 2024. Disponível em: <https://glasswing.vc/blog/research/the-history-of-artificial-intelligence/>.
- SARCEVIC, Tanja et al. U Can't Gen This? A Survey of Intellectual Property Protection Methods for Data in Generative AI, 2024. DOI: 10.48550/arXiv.2406.15386. Disponível em: <https://arxiv.org/abs/2406.15386>.
- SINGH, Kamalmeet. **A Brief History of Generative AI**. [S.l.: s.n.], 2024. Disponível em: <https://medium.com/@kamalmeet/a-brief-history-of-generative-ai-e91a95d5604c>.
- TEAM, Tencent Hunyuan3D. **Hunyuan3D 2.0: Scaling Diffusion Models for High Resolution Textured 3D Assets Generation**. [S.l.: s.n.], 2025. Disponível em: <https://github.com/Tencent-Hunyuan/Hunyuan3D-2>.
- TERO KARRAS MIIKA AITTALA, Timo Aila; LAINE, Samuli. **Elucidating the Design Space of Diffusion-Based Generative Models**. [S.l.: s.n.], 2022. Disponível em: [https://proceedings.neurips.cc/paper\\_files/paper/2022/hash/a98846e9d9cc01cfb87eb694d946ce6b-Abstract-Conference.html](https://proceedings.neurips.cc/paper_files/paper/2022/hash/a98846e9d9cc01cfb87eb694d946ce6b-Abstract-Conference.html).
- WALKER, Stephen M. **Diffusion Models**. [S.l.: s.n.], 2023. Disponível em: <https://klu.ai/glossary/diffusion-models>.
- ZHANG, Lvmin; RAO, Anyi; AGRAWALA, Maneesh. Adding Conditional Control to Text-to-Image Diffusion Models, 2023. Disponível em: <https://arxiv.org/pdf/2302.05543>.

## APÊNDICE A – RELATÓRIO TÉCNICO

O relatório técnico completo da Revisão Sistemática sobre o uso de Inteligência Artificial Gerativa na modelagem 3D para Desenvolvimento de Jogos está disponível online no site do Instituto de Pesquisa, podendo ser acessado pelo ResearchGate: [https://www.researchgate.net/publication/381733374\\_Revisao\\_Sistematica\\_do\\_Uso\\_de\\_Inteligencia\\_Artificial\\_Gerativa\\_na\\_modelagem\\_3D\\_para\\_Desenvolvimento\\_de\\_Jogos](https://www.researchgate.net/publication/381733374_Revisao_Sistematica_do_Uso_de_Inteligencia_Artificial_Gerativa_na_modelagem_3D_para_Desenvolvimento_de_Jogos).



**INCoD**

Instituto Nacional para Convergência Digital



UNIVERSIDADE FEDERAL  
DE SANTA CATARINA

ISSN 2236-5281

Technical Report

**INCoD/LAPIX.02.2024.P**

DOI: 10.13140/RG.2.2.32290.03520

# **Revisão Sistemática do Uso de Inteligência Artificial Gerativa na modelagem 3D para Desenvolvimento de Jogos**

**Autores:**

Emily Sayuri Kiba

Aldo von Wangenheim

Version 1.0

Status: Final

Distribution: External

**JUNHO - 2024**

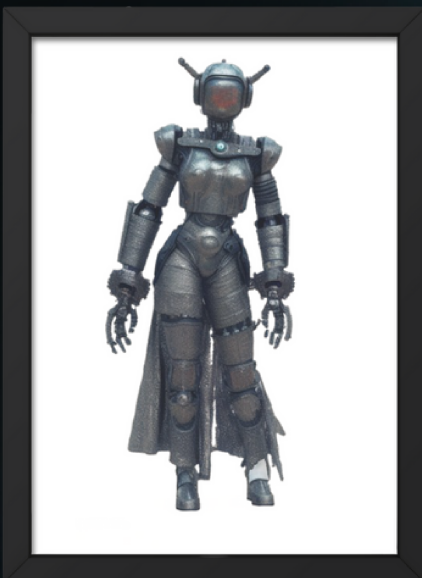
## APÊNDICE B – MANUAL PRÁTICO

O manual prático completo está disponível online no site do Instituto de Pesquisa, podendo ser acessado através do ResearchGate: [https://www.researchgate.net/publication/396709256\\_MANUAL\\_DE\\_IA\\_GERATIVA\\_PARA\\_MODELAGEM\\_DE\\_PERSONAGEM\\_3D\\_-\\_GENERATIVE\\_AI\\_MANUAL\\_FOR\\_3D\\_CHARACTER\\_MODELLING](https://www.researchgate.net/publication/396709256_MANUAL_DE_IA_GERATIVA_PARA_MODELAGEM_DE_PERSONAGEM_3D_-_GENERATIVE_AI_MANUAL_FOR_3D_CHARACTER_MODELLING).

# MANUAL DE IA GERATIVA PARA MODELAGEM DE PERSONAGEM 3D

## *GENERATIVE AI MANUAL FOR 3D CHARACTER MODELLING*

EMILY SAYURI KIBA  
ALDO VON WANGENHEIM



Version 1.0  
Status: Final  
Distribution: External  
**OCTOBER - 2025**

## **APÊNDICE C – PROTÓTIPO DE JOGO EM UNREAL ENGINE**

O protótipo do jogo desenvolvido no Unreal Engine 5, incluindo os personagens gerados por IA, animações e cenário, está disponível em: [https://drive.google.com/file/d/1pQfHpvgdQbB44myC\\_YYXc\\_CcR6K0d76X/view?usp=sharing](https://drive.google.com/file/d/1pQfHpvgdQbB44myC_YYXc_CcR6K0d76X/view?usp=sharing).

**APÊNDICE D – ARTIGO**

Artigo no formato especificado pela Sociedade Brasileira de Computação (SBC).

# Manual Prático de Uso de Inteligência Artificial Gerativa na Criação de Modelos 3D para Jogos Digitais

Emily Sayuri Kiba<sup>1</sup>, Aldo von Wangenheim<sup>1</sup>

<sup>1</sup>Departamento de Informática e Estatística  
Centro Tecnológico – Universidade Federal de Santa Catarina (UFSC)

**Abstract.** *This work investigates the use of Generative Artificial Intelligence, particularly the Stable Diffusion model, applied to 3D modeling and its integration into digital game development. The objective was to develop a practical guide that demonstrates how these techniques can support the generation of ideas, visual concepts, meshes, and textures for three-dimensional models. The methodology involved hands-on experimentation with different workflows using Stable Diffusion and auxiliary tools, evaluating the quality of the results and their applicability in the 3D creation process. As a result, a practical manual was produced to guide beginners in the agile creation of 3D assets, along with the development of a game prototype in Unreal Engine 5 using AI-generated elements. The results indicate that generative AI can significantly reduce production time and improve accessibility to modeling processes, especially for novice users. The contributions of this study include the systematization of an accessible workflow and the demonstration of AI's potential as a support tool for 3D modeling in games.*

**Keywords:** *Generative Artificial Intelligence. Stable Diffusion. 3D Modeling. Game Engine.*

**Resumo.** *Este trabalho investiga o uso de Inteligência Artificial (IA) Gerativa, em especial o modelo Stable Diffusion, aplicada à modelagem tridimensional (3D) e sua integração no desenvolvimento de jogos digitais. O objetivo foi elaborar um manual prático que demonstra como essas técnicas podem apoiar a geração de ideias, conceitos visuais, malhas e texturas para modelos tridimensionais. A metodologia envolveu experimentação prática com diferentes fluxos de trabalho utilizando Stable Diffusion e ferramentas auxiliares, avaliando a qualidade dos resultados e sua aplicabilidade no processo de criação 3D. Como resultado, foi produzido um manual prático que orienta iniciantes na criação ágil de ativos 3D, além da construção de um protótipo de jogo na Unreal Engine 5 utilizando elementos gerados por IA. Os resultados indicam que a IA gerativa pode reduzir significativamente o tempo de produção e facilitar o acesso a processos de modelagem, especialmente para usuários iniciantes. As contribuições deste estudo incluem a sistematização de um fluxo de trabalho acessível e a demonstração do potencial da IA como ferramenta de apoio à modelagem 3D em jogos.*

**Palavras-chave:** *Inteligência Artificial Gerativa. Stable Diffusion. Modelagem 3D. Game Engine.*

## 1. Introdução

A modelagem 3D é essencial na criação de jogos digitais, animações e experiências imersivas, mas o desenvolvimento de ativos 3D exige tempo, habilidades técnicas e conheci-

mento em arte digital. Com o crescente interesse de desenvolvedores e entusiastas sem formação em computação gráfica, surge a necessidade de ferramentas que facilitem o processo criativo e acelerem a produção de conteúdo.

Recentemente, modelos de IA gerativa, como o Stable Diffusion, popularizaram-se por permitir a geração de imagens e texturas a partir de descrições textuais ou imagens de referência, oferecendo recursos como inpainting, upscaling, ControlNet e *Low-Rank Adaptation* (LoRA). Essas técnicas ajudaram a popularizar a IA gerativa, tornando-a poderosa para design conceitual em qualquer disciplina que exija criatividade visual, incluindo estágios iniciais de projetos arquitetônicos, como ideação, esboço e modelagem (Ploennigs; Berger, 2023). As capacidades gerativas dessas ferramentas provavelmente alteraram fundamentalmente os processos criativos, influenciando a forma como ideias são formuladas e transformadas em produtos (Epstein; Hertzmann, 2023).

Apesar de permitir prototipagem rápida e acelerar o design, a IA gerativa apresenta curva de aprendizado significativa, exigindo compreensão de parâmetros e entradas para gerar resultados desejados. Diante disso, este trabalho propõe a aplicação dessas ferramentas na modelagem 3D para jogos digitais, com a elaboração de um manual prático que sistematiza fluxos de trabalho com Stable Diffusion e suas interfaces Automatic1111 e ComfyUI, e a criação de um protótipo de jogo na Unreal Engine 5 que demonstra a integração de ativos e personagens gerados por IA. Os resultados indicam que tais ferramentas facilitam a modelagem 3D para iniciantes, apoiam o processo criativo e promovem eficiência, democratização do acesso e inovação na criação de conteúdo digital.

## 1.1. Objetivos

O objetivo deste trabalho é investigar o uso de ferramentas de IA gerativa na criação de imagens e modelos 3D para apoiar o desenvolvimento visual de jogos digitais. Busca-se demonstrar como essas ferramentas podem agilizar a concepção de personagens, objetos e cenários, além de gerar modelos 3D a partir de imagens, consolidando os procedimentos em um manual prático voltado a iniciantes.

Os objetivos específicos incluem explorar conceitos de IA gerativa, investigar interfaces como Automatic1111 e ComfyUI, aplicar técnicas de Image-to-3D, desenvolver o manual prático, avaliar o impacto no processo criativo e criar um protótipo de jogo utilizando os ativos gerados.

## 2. Fundamentação Teórica e Ferramentas Utilizadas

O presente estudo utiliza o Stable Diffusion, um modelo de difusão latente, que permite gerar imagens de alta qualidade a partir de descrições textuais ou imagens de referência. Os modelos de difusão operam adicionando ruído progressivamente a uma imagem e, em seguida, removendo-o iterativamente, o que possibilita a criação de conteúdos visuais detalhados e controláveis. Eles são considerados uma classe de modelos de IA gerativa capazes de gerar imagens de alta qualidade, ao contrário de outros modelos, como *Generative Adversarial Network* (GAN) e *Variational Autoencoder* (VAE), que apresentam dificuldades para produzir imagens detalhadas em alta resolução (Ahirwar, 2023).

Para a geração de imagens e a criação de referências para modelos 3D, foram aplicadas técnicas como:

- **Text-to-Image:** geração de imagens a partir de *prompts* textuais.

- **Image-to-Image**: transformação de uma imagem inicial guiada por prompt textual, mantendo a estrutura da imagem original.
- **Inpainting**: modificação localizada de imagens, permitindo apagar ou substituir partes específicas.
- **ControlNet**: adição de controles estruturais, como esboços ou poses, para guiar a geração.
- **LoRA**: adaptação de modelos de grande escala para estilos específicos, com menor custo computacional.

Para manipular o Stable Diffusion de forma prática, foram utilizadas duas interfaces gráficas:

- **Automatic1111**: interface popular, intuitiva e com amplo suporte a extensões, adequada para experimentação geral.
- **ComfyUI**: interface baseada em nós, que permite maior controle sobre cada etapa do fluxo de geração, sendo eficiente para criar mapas de texturas e ativos 3D complexos, com ganho de performance em workflows mais avançados.

Nos últimos anos surgiram também ferramentas voltadas diretamente para modelagem 3D baseada em IA, como Meshy AI e Hunyuan 3D, que permitem gerar malhas 3D a partir de imagens ou descrições textuais. Neste estudo, essas ferramentas foram utilizadas para apoiar a criação de modelos tridimensionais dentro do fluxo de produção de jogos digitais, permitindo a geração e o ajuste de ativos 3D de forma prática e ágil.

### 3. Metodologia e Desenvolvimento

Este estudo propôs um fluxo de trabalho para utilização de IA gerativa na criação de ativos 3D para jogos digitais, consolidado em um manual prático que abrange desde a configuração do ambiente até a integração dos ativos no protótipo de jogo. O manual detalha a instalação e configuração do Stable Diffusion, utilizando as interfaces Automatic1111 e ComfyUI, destacando suas principais diferenças em termos de usabilidade e desempenho.

#### 3.1. Geração de imagens

Na etapa de geração de imagens, o manual detalha os parâmetros mais importantes que influenciam o resultado final, como número de steps, *Classifier-Free Guidance* (CFG) *scale*, tamanho da imagem, *denoise strength* e escolha de modelos. São fornecidos exemplos visuais comparativos para demonstrar como cada parâmetro altera o resultado. O manual também indica em cada interface onde esses parâmetros podem ser ajustados, oferecendo maior controle e precisão sobre a geração de imagens.

Além disso, são fornecidos fluxos de trabalho para geração de mapas de texturas *Physically Based Rendering* (PBR), essenciais para materiais 3D realistas, permitindo que o usuário reproduza a criação de texturas consistentes com os modelos gerados.

Para ComfyUI, todos esses fluxos estão disponibilizados em arquivos JSON, correspondentes a cada técnica, garantindo que o usuário possa reproduzir exatamente os processos demonstrados no manual. A Figura 1 apresenta um exemplo do workflow de Image-to-Image no ComfyUI, evidenciando o encadeamento de nós e configurações utilizadas para gerar imagens refinadas a partir de referências existentes.

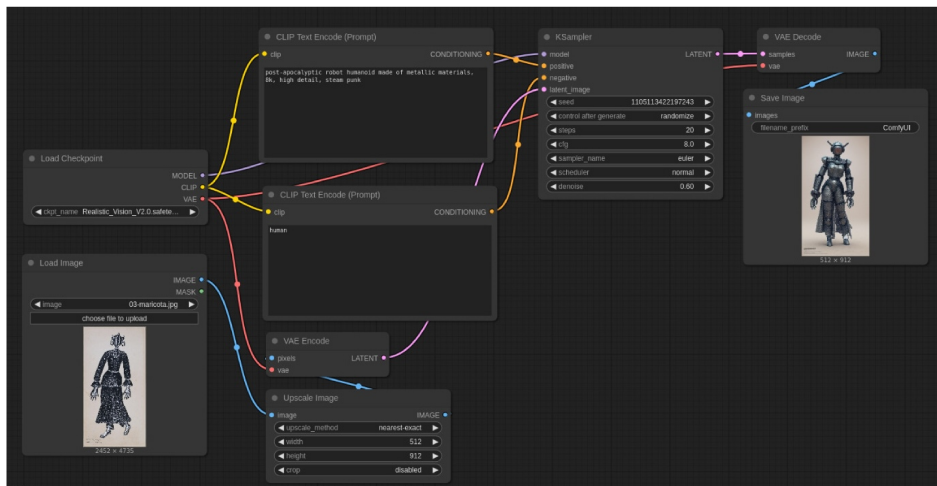


Figura 1. Workflow de técnica image-to-image. Fonte: autoria própria

### 3.2. Geração de modelos 3D

As imagens geradas na etapa anterior serviram como base para a criação de modelos 3D, realizada com ferramentas de Image-to-3D — Hunyuan 3D 2.0, Hunyuan 3D 2.5 e Meshy AI. No manual foca especialmente no Hunyuan 3D 2.0, desenvolvido pela Tencent, por permitir a execução local sem limites de uso. Esta ferramenta possibilita criar malhas 3D otimizadas e gerar mapas de textura realistas a partir de uma única imagem 2D. Segundo o artigo publicado pela *Towards Deep Learning*, o Hunyuan 3D 2.0 representa um avanço significativo ao democratizar uma tecnologia antes acessível apenas a profissionais especializados, oferecendo aplicabilidade prática para áreas como jogos, cinema, animação e impressão 3D, além de reduzir custos na produção de ativos 3D.

O fluxo de trabalho utilizado para gerar modelos 3D a partir de uma imagem com o Hunyuan 3D 2.0 é ilustrado na Figura 2. Este workflow demonstra as etapas principais, desde a importação da imagem gerada pela IA até a produção do modelo 3D final, incluindo ajustes manuais quando necessários para correção de topologia, geometria ou texturas.

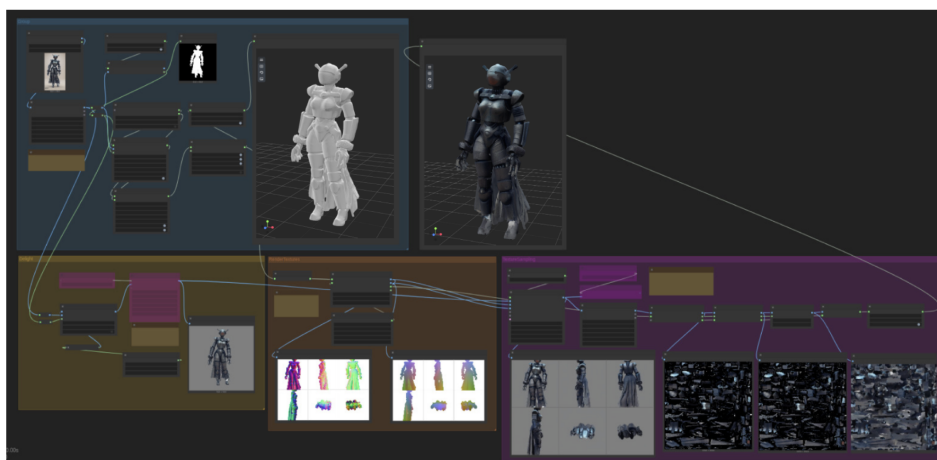


Figura 2. Workflow de geração de modelo 3D a partir de imagem utilizando Hunyuan 3D 2.0. Fonte: autoria própria

Embora todas as ferramentas permitam gerar modelos 3D a partir de imagens, foi identificada variação na qualidade final, especialmente em topologia, fidelidade da textura e precisão estrutural. Essas diferenças podem exigir ajustes manuais no Blender para garantir geometria, topologia e texturização adequadas. Os principais problemas identificados e suas soluções estão resumidos na Tabela 1.

<b>Problema</b>	<b>Correção / Solução</b>
Malha colada	Ajuste manual no Blender (Knife bisect ou deletar face diretamente)
Texturas incorretas ou incompletas	Aplicação e correção manual de texturas, retoque visual no Blender
Shader mal configurado	Ajuste de parâmetros de Shader no Blender

**Tabela 1. Principais problemas em modelos 3D gerados por IA e correções aplicadas. Fonte: autoria própria**

### 3.3. Aplicação de Esqueleto

Posteriormente, os modelos 3D passam pela etapa de aplicação de esqueleto (*rigging*), essencial para possibilitar animações e a integração no protótipo de jogo na Unreal Engine 5. O manual aborda duas abordagens principais para rigging:

O rigging manual exige precisão no alinhamento do esqueleto com a malha do personagem, garantindo que os ossos e vértices estejam corretamente associados e prevenindo deformações durante a animação. O processo envolve:

- **Ajuste do Esqueleto:** o esqueleto é posicionado de acordo com a anatomia do modelo.
- **Conexão com a Malha:** associação da malha ao esqueleto, geralmente com pesos automáticos de influência.
- **Correção de Influência de Pesos (*Weight Painting*):** ajustes manuais quando a associação automática não é precisa, utilizando a ferramenta de pintura de pesos no Blender.
- **Exportação:** o modelo rigged é exportado no formato `.fbx` para utilização na Unreal Engine 5.

O rigging automático proporciona maior agilidade, mas ainda pode exigir ajustes manuais, especialmente dependendo da topologia da malha, que pode impedir a identificação correta de membros e a aplicação automática do esqueleto:

- **Plataformas de IA:** Hunyuan 3D 2.5 e Meshy AI permitem gerar modelos 3D já com esqueleto aplicado.
- **Blender Rigify:** *add-on* que realiza *auto-rigging* a partir de um metarig, permitindo ajustes finos.
- **Mixamo:** ferramenta online para criação rápida de esqueletos humanoides com animações pré-definidas.

### 3.4. Criação de Jogo

O protótipo desenvolvido integrou os modelos 3D gerados por IA em um ambiente interativo na Unreal Engine 5, demonstrando a aplicação prática dos ativos digitais. O sistema de regras e interações foi implementado utilizando *Blueprints*, com o objetivo de apresentar a funcionalidade dos modelos em um contexto de jogo.

A mecânica principal consiste na coleta de gemas para acumular pontos, sendo que gemas menores contribuem com pontuações parciais, enquanto a gema final vale 100 pontos. O acesso à gema de maior valor é condicionado à ativação de duas plataformas interativas, que reduzem a pontuação do jogador como custo de progressão. Uma vez que ambas as plataformas são ativadas, a barreira que bloqueia a gema final é removida, permitindo que o jogador conclua o objetivo do jogo.

Para aumentar a imersão, foram aplicadas técnicas de texturização avançada e inserção de elementos de cenário, como vegetação e objetos adicionais, utilizando ferramentas como *Foliage Tool*, *tiling*, *texture bombing* e *blending*.

## 4. Resultados

### 4.1. Resultados da Geração dos Modelos 3D

A Figura 3 apresenta um exemplo representativo do fluxo utilizado neste trabalho. A imagem original da personagem Maricota, de Franklin Cascaes, foi utilizada como entrada para a geração de uma variação estilizada por IA. Em seguida, essa imagem processada serviu como base para a etapa de Image-to-3D, resultando no modelo 3D exibido na figura.

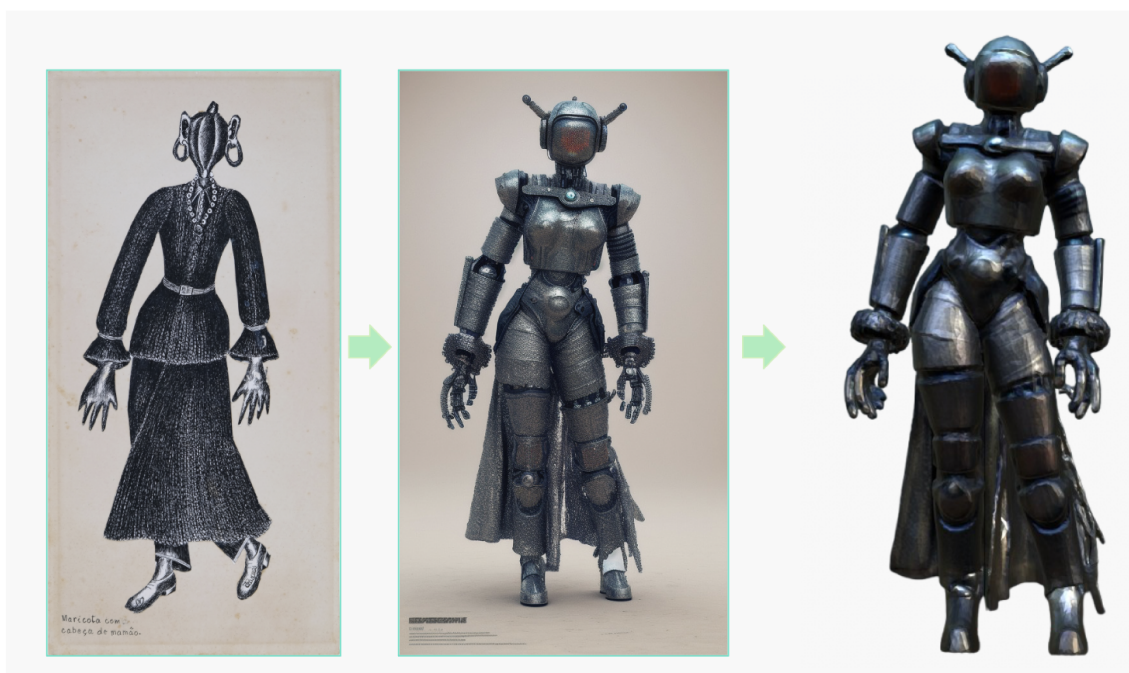


Figura 3. Processo de criação de personagem 3D. Fonte: autoria própria

## 4.2. Comparação de modelos que geram malha 3D

Foram utilizadas diferentes ferramentas de Image-to-3D — Hunyuan 3D 2.0, Hunyuan 3D 2.5 e Meshy AI — com a mesma imagem de entrada. Como mostrado na Figura 4, cada ferramenta apresentou características visuais e estruturais distintas. A Tabela 2 sintetiza as principais diferenças observadas, incluindo qualidade da malha e textura, limites de uso e tipo de execução.

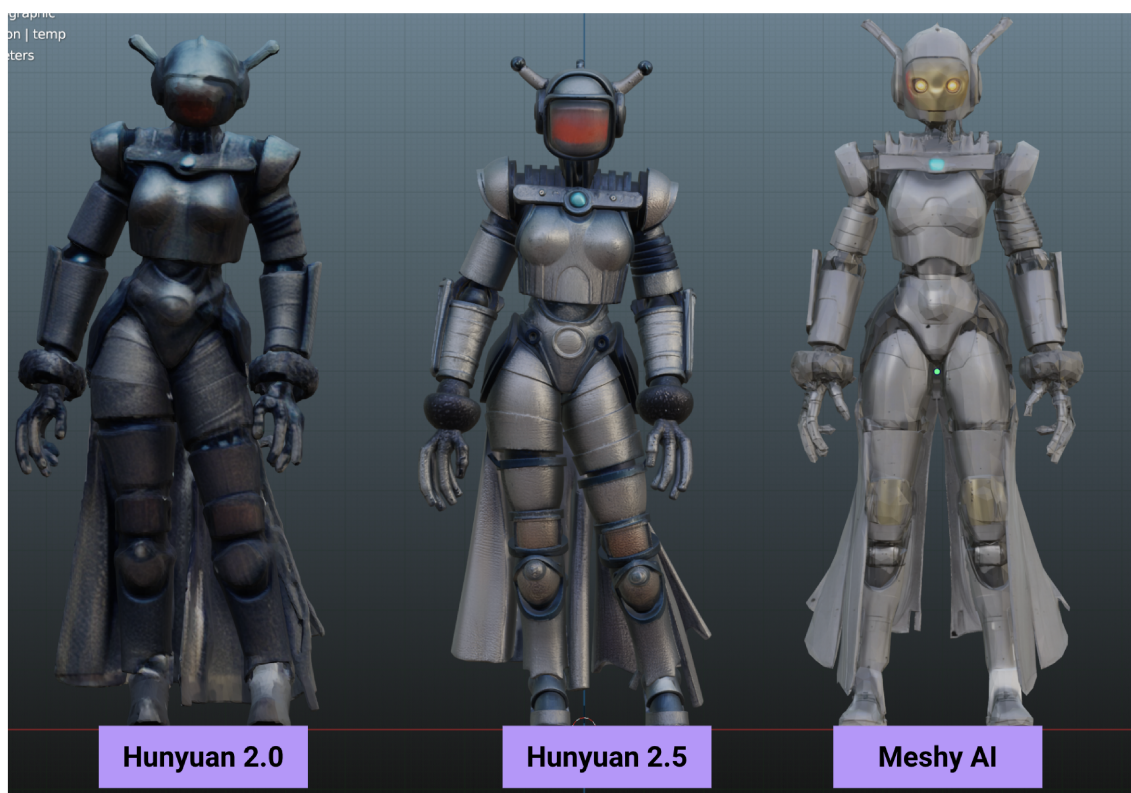


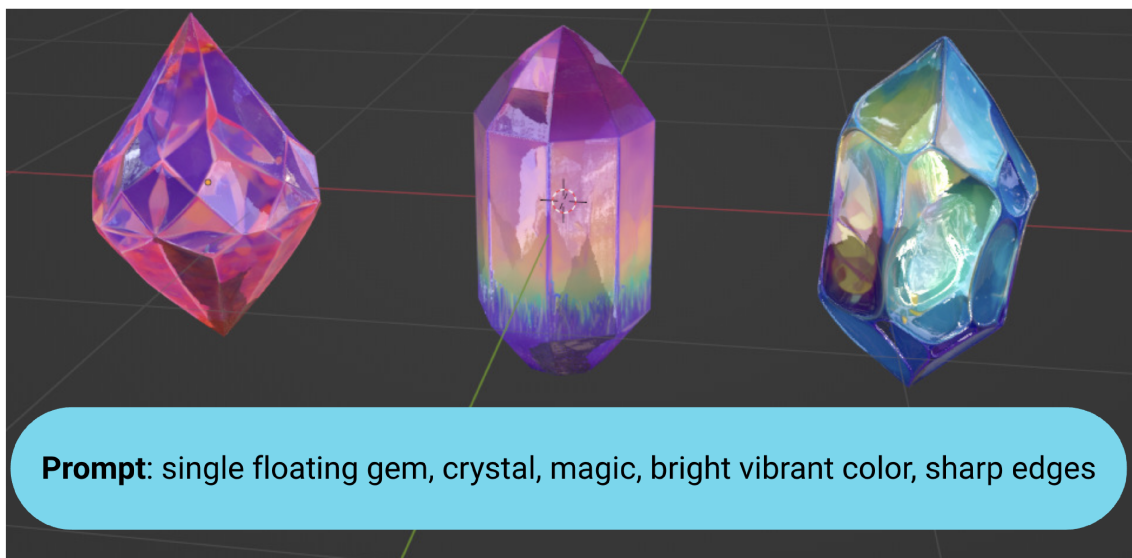
Figura 4. Comparação entre resultados gerados por modelos de IA gerativa diferentes. Fonte: autoria própria

Ferramenta	Qualidade da Malha/Textura	Execução	Limites de Uso
Hunyuan 3D 2.0	Média, pequenas falhas	Local	Uso ilimitado
Hunyuan 3D 2.5	Alta, qualidade muito boa	Plataforma Online	20 gerações/dia
Meshy AI	Boa, mas menos fiéis à entrada	Plataforma Online	20 gerações/mês

Tabela 2. Comparação entre ferramentas de geração de modelos 3D a partir de imagens. Fonte: autoria própria

## 4.3. Ativos e Integração no Jogo

Os ativos como gemas, plataformas e grades de madeira foram criados com o Hunyuan 3D 2.5 via Text-to-3D, demonstrando a viabilidade de gerar objetos jogáveis com auxílio de IA. A Figura 5 apresenta as gemas geradas partir do prompt textual apresentado.



**Figura 5. Gemas criadas por Hunyuan 3D 2.5. Fonte: autoria própria**

O protótipo desenvolvido utilizou como cenário a Fortaleza de Anhatomirim, fornecida pelo professor Flávio Andaló, adaptada para o jogo com elementos adicionais e substituição de texturas por versões geradas por IA. A Figura 6 e 7 apresentam o protótipo do jogo, destacando a aplicação dos ativos e texturas geradas por IA, bem como a composição do cenário com vegetação e elementos interativos. Essa demonstração evidencia a viabilidade do uso de IA gerativa para acelerar a criação de conteúdos jogáveis, mantendo controle sobre a estética e a funcionalidade do ambiente.



**Figura 6. Ambiente do jogo: Ilha do Anhatomirim. Fonte: autoria própria**



**Figura 7. Entrada da Ilha do Anhatomirim. Fonte: autoria própria**

## **5. Conclusão**

Este trabalho analisou a aplicação prática de ferramentas de IA gerativa na criação de ativos visuais para jogos digitais, destacando sua capacidade de agilizar a prototipagem e aumentar a diversidade estética. A utilização de diferentes modelos de IA, bem como técnicas como Inpainting, ControlNet e LoRA, permitiu maior controle sobre composição, poses, estilos e correções localizadas das imagens. A experimentação demonstrou que usuários sem experiência prévia em modelagem 3D podem transformar conceitos visuais em ativos 3D funcionalmente integráveis a um protótipo de jogo.

Entretanto, a geração automática ainda apresenta limitações significativas. O controle sobre o resultado final é parcial, e os modelos 3D frequentemente requerem ajustes manuais em softwares como Blender para corrigir topologia, geometria irregular ou superfícies incompletas. Assim, o julgamento humano permanece essencial para validar, selecionar e refinar os resultados, garantindo coerência estética e precisão estrutural.

O estudo também abordou aspectos éticos e legais relacionados ao uso de IA gerativa, ressaltando riscos de violação de direitos autorais e mimese de estilos de artistas presentes nos dados de treinamento. Além disso, designers têmem que o conteúdo gerado possa resultar em plágio não intencional ou disputas jurídicas, tornando essencial que estúdios de jogos implementem políticas claras sobre o uso da IA e os limites da autoria criativa (Alharthi, 2025). Para uso responsável, recomenda-se conhecer a origem dos modelos e datasets, documentar o processo, manter transparência sobre a participação da IA e adotar práticas que respeitem a diversidade, acompanhando legislações emergentes.

Em síntese, a IA gerativa deve ser entendida como uma ferramenta de apoio à criatividade humana. Sua integração eficiente acelera a produção de conteúdos jogáveis, amplia possibilidades artísticas e democratiza o acesso à modelagem 3D, mas depende de supervisão, reflexão crítica e responsabilidade ética e legal do usuário. O uso pleno e sustentável da IA resulta do equilíbrio entre inovação tecnológica e julgamento humano.

## Referências

AHIRWAR, Kailash. **A Very Short Introduction to Diffusion Models**. [S. l.: s. n.], 2023. Disponível em: <https://kailashahirwar.medium.com/a-very-short-introduction-to-diffusion-models-a84235e4e9ae>.

ALHARTHI, Sultan A. **Generative AI in Game Design: Enhancing Creativity or Constraining Innovation?**, 2025. DOI: 10.3390/jintelligence13060060. Disponível em: <https://www.mdpi.com/2079-3200/13/6/60>.

EPSTEIN, Ziv; HERTZMANN, Aaron. **Art and the science of generative AI**. **Science**, jun. 2023. DOI: 10.1126/science.adh4451. Disponível em: <https://doi.org/10.1126/science.adh4451>.

PLOENNIGS, Jonas; BERGER, Matthias. **AI art in architecture**. **AI in Civil Engineering**, v. 2, n. 8, 2023. DOI: 10.1007/s43503-023-00018-y. Disponível em: <https://doi.org/10.1007/s43503-023-00018-y>.