

# Inteligência Artificial gerativa Para Criação de Conteúdo em Jogos Digitais

Matheus B. Torres

<sup>1</sup>Centro de Informática – Universidade Federal de Pernambuco (UFPE)  
Caixa Postal 7851. 50732-970 – Recife – PE – Brazil

mbmt@cin.ufpe.br

**Abstract.** *The creation of levels is a significant challenge in the field of digital game development, requiring meticulous and creative work for challenging games that captivate the player. However, generative AI techniques can assist in this process. By using adversarial learning models, it's possible to generate original content autonomously, providing an innovative solution for level creation. In this work, we will explore the current state of generative AI in level generation for digital games, analyzing a set of articles that delve into the use of these techniques.*

**Resumo.** *A criação de níveis é um grande desafio no desenvolvimento de campo dos jogos digitais, exigindo um trabalho minucioso e criativo para jogos desafiadores e que cativam o jogador. No entanto, técnicas de IA gerativa podem auxiliar nesse processo. Com o uso de modelos de aprendizagem adversarial é possível gerar conteúdo original de forma autônoma, oferecendo uma solução inovadora para a criação de níveis. Neste trabalho, exploraremos o estado atual da IA gerativa na geração de níveis em jogos digitais, analisando um conjunto de artigos que exploram a utilização dessas técnicas.*

## 1. Introdução

A criação de conteúdo para jogos digitais é uma tarefa complexa que requer tempo, habilidade e criatividade por parte dos desenvolvedores e artistas envolvidos. Para isso foram criadas técnicas de geração procedural de conteúdo. Uma das atividades que carecem dessa criatividade é o design de layouts de níveis, muito já se avançou nessa área, permitindo que jogos tenham seus níveis inteiramente gerados de forma procedural com ou sem a assistência de um agente humano, normalmente a produção de design de níveis é exercida pelo profissional denominado game designer.

No entanto, com os avanços na área de inteligência artificial (IA), surge uma abordagem que pode aprimorar esse processo: a IA gerativa. Com o uso de modelos GAN é possível de criar conteúdo original de forma autônoma a partir de um conjunto de níveis existentes, oferecendo uma nova perspectiva para a criação de níveis em jogos digitais. Neste artigo, exploraremos o estado atual da IA gerativa geração de níveis de jogos digitais, examinando seus benefícios, desafios e potencial para transformar a forma como a geração de níveis é concebida. Além disso, apresentaremos exemplos concretos e estudos de caso que ilustram como essa tecnologia está impulsionando a criatividade, a eficiência e a qualidade na produção de elementos para jogos digitais gerando design de níveis. Por fim, refletiremos sobre as implicações futuras desse avanço tecnológico e o impacto que

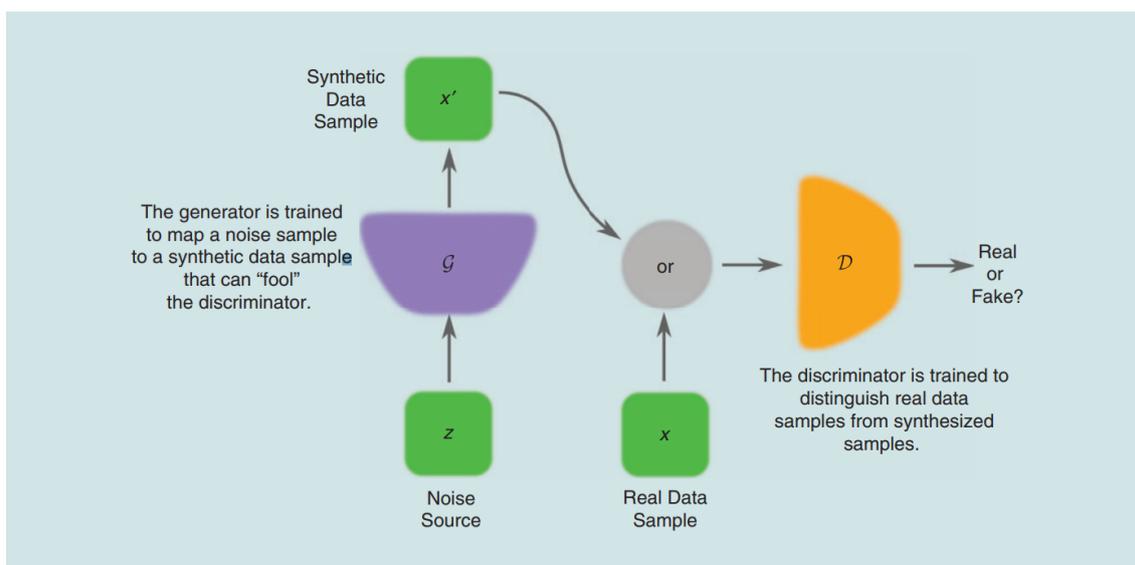
a IA gerativa pode ter na indústria de jogos como um todo. Este artigo tem como objetivo fornecer uma visão abrangente e atualizada sobre a criação de níveis para jogos digitais com o uso da IA gerativa, além de estimulando a discussão sobre o potencial dessa tecnologia e suas implicações para o futuro dos jogos digitais.

O artigo está dividido nas seguintes seções: Modelos, onde serão abordados os principais modelos utilizados nos estudos. Metodologia, onde será explicada a metodologia utilizada neste estudo. Resultados, onde serão apresentados os resultados obtidos nos estudos abordados assim como as técnicas abordadas e avaliação feitas. Conclusão e Discussão, onde abordaremos revisaremos os resultados e refletimos sobre os resultados obtidos. Trabalhos Futuros onde analisamos que próximos passos podem ser tomados na área.

## 2. Modelos

Rede Adversarial Generative(GAN) é um tipo de modelo de aprendizado que consiste em 2 principais componentes, o gerador e o discriminador. Em síntese, nesses modelos o discriminador é responsável por identificar se uma dada entrada é real ou se foi gerada pelo gerador, enquanto o gerador é responsável por gerar elementos do conjunto de entrada do discriminador. Para o treinamento do modelo é realizado um processo de treinamento adversarial onde o discriminador melhora sua capacidade de identificar dados do conjunto de treinamento, e o gerador melhora sua capacidade de gerar entradas capazes de enganar o discriminador. Após o treinamento o discriminador é descartado e o gerador pode ser utilizado para criar dados semelhantes aos reais. Uma ilustração da arquitetura do modelo GAN tradicional como descrita pode ser observada na figura 1.

**Figura 1. Ilustração de treinamento GAN.**



Fonte: [Creswell et al. 2018].

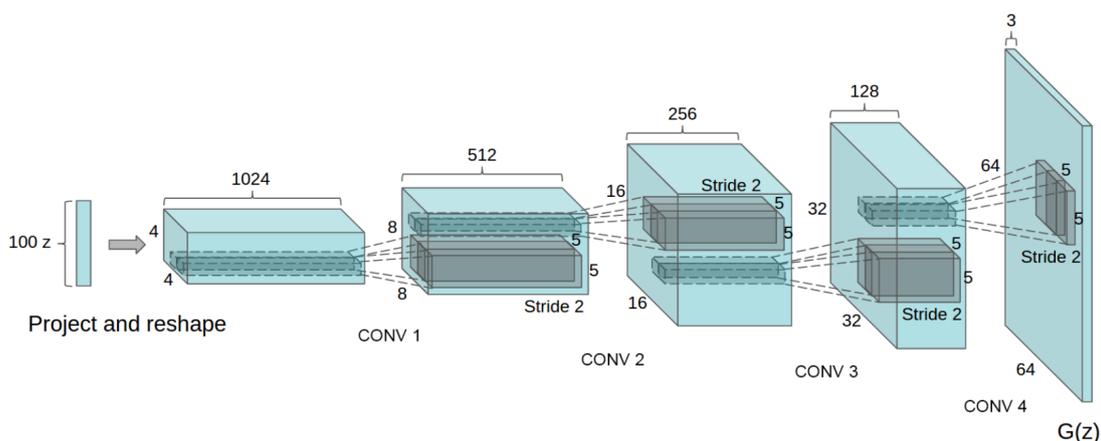
Essa classe de modelos já é aplicada com diversas finalidades como a geração de texto ou imagens e existe uma variedade de arquiteturas que podem ser aplicadas para o fim que buscamos neste artigo. Diversas arquiteturas GAN foram abordadas nos arti-

gos, sendo a mais popular delas o DCGAN, uma arquitetura particularmente boa para a geração de imagens.

## 2.1. DCGAN

O DCGAN (Deep Convolutional Generative Adversarial Network) é uma arquitetura de rede neural usada para treinar modelos generativos. Ele se caracteriza por incorporar camadas convolucionais profundas tanto no gerador quanto no discriminador. Além de normalização em lote, técnica do processo de treinamento da rede que opera normalizando as ativações em mini-lotes de dados, o que reduz a instabilidade e acelera a convergência. É utilizada também a função de ativação unidade linear retificada (ReLU), que opera retornando 0 para valores negativos e o próprio valor de entrada para outros valores, permitindo resultados de alta qualidade. O DCGAN é amplamente empregado em tarefas de síntese de imagens e tem sido fundamental para o avanço das aplicações de geração de imagens por meio de aprendizado de máquina [Radford et al. 2015]. Um exemplo de arquitetura DCGAN utilizada para geração de imagens pode ser observada na figura 2.

Figura 2. Exemplo de arquitetura DCGAN utilizada para gerar imagens.



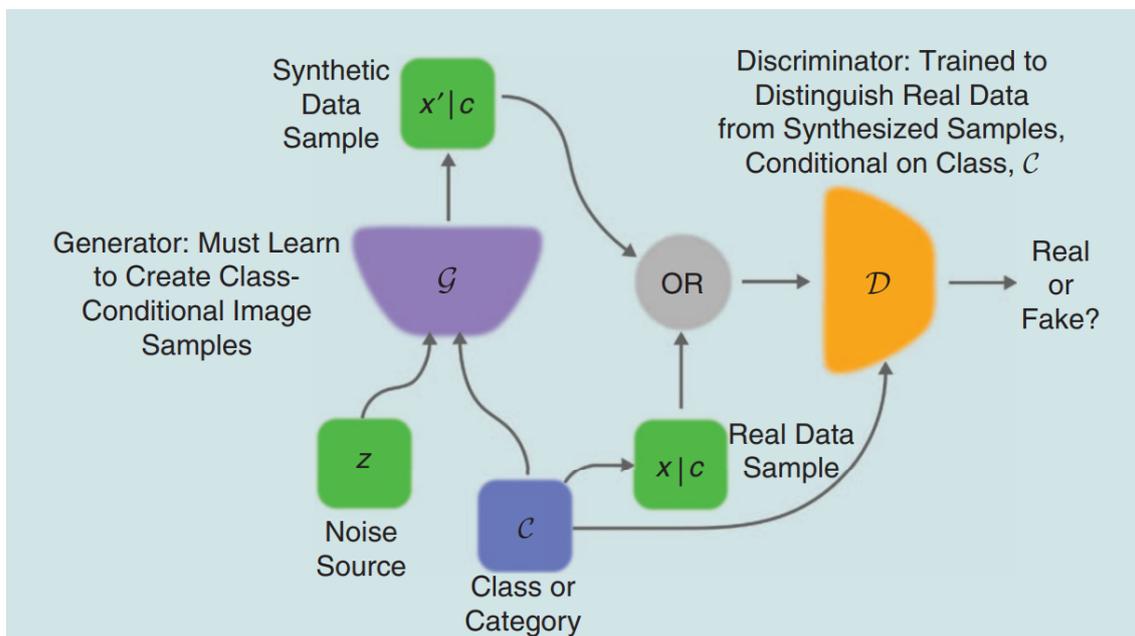
Fonte: [Radford et al. 2015].

## 2.2. GAN condicional

Um ponto que foi recorrente em alguns dos estudos foi o uso de modelos GAN condicionais (CGAN) e não condicionais. Os modelos condicionais recebem uma forma de entrada que será usada para reduzir o conjunto de resultados que pode ser produzido, esse tipo de modelo pode ser utilizado por exemplo para produzir imagens a partir de entradas de texto, servindo como um 'guia' para o gerador criar resultados específicos. Dessa forma os modelos CGAN funcionam de forma similar aos modelos não condicionais, entretanto o gerador recebe um vetor adicional que condiciona a saída esperada e o discriminador também recebe esse vetor para então avaliar não somente se os dados gerados são falsos, mas também se eles satisfazem as condições esperadas [Gao et al. 2021].

Uma ilustração da arquitetura CGAN pode ser observada na figura 3, diferente do observado na figura 1 o gerador e discriminador recebem um parâmetro a mais que corresponde à classe dos dados que podem ser reais ou gerados.

Figura 3. Arquitetura de GAN condicional.



Fonte: [Creswell et al. 2018].

No caso da geração de conteúdo procedural para jogos o CGAN tem diversas aplicações que podem tanto auxiliar a obter características desejadas em níveis quanto ser utilizado para manter uma coesão entre os elementos gerados. Em se tratando de níveis criados por redes adversariais não existe garantia de que seja possível que o jogador complete ou que elementos estejam dispostos de forma incorreta, com o uso de redes condicionais é possível induzir ou condicionar os resultados gerados, melhorando os resultados obtidos.

### 2.3. MultiGan

No contexto do MultiGAN, várias dessas duplas gerador-discriminador são utilizadas em conjunto, cada uma responsável por gerar uma parte específica de um resultado complexo, como uma imagem, música ou texto. Essas redes colaboram entre si para criar uma saída integrada e mais rica, permitindo a geração de conteúdo mais realista e diversificado do que uma única GAN poderia alcançar. O MultiGAN é frequentemente utilizado em tarefas que exigem a síntese de informações multifacetadas e complexas, como a criação de arte gerativa, geração de texto natural e design de produtos personalizados [Capps and Schrum 2021].

## 3. Metodologia

Neste artigo foi feito um mapeamento sistemático, com o objetivo de responder qual o estado atual do uso de IA gerativa para a criação de níveis em jogos digitais. Primeiramente foi formado um conjunto de artigos primários que contemplem o uso de modelos de Generative AI para a geração de níveis de jogos digitais. O conjunto possui artigos escritos em diversos países, entre 2018 e 2023 sendo a maior parte deles do Reino Unido e Estados Unidos, todos em língua inglesa.

Os artigos foram buscados com a string de busca '*generative AI level generation*', os artigos foram buscados utilizando a ferramenta *Google Scholar*, os critérios de inclusão foram:

- Estudos primários
- Estudos feitos nos últimos 5 anos
- Estudos em inglês ou português
- Estudos Revisados por pares
- Estudos que abordem geração de níveis com inteligência artificial

Os critérios de exclusão utilizados foram:

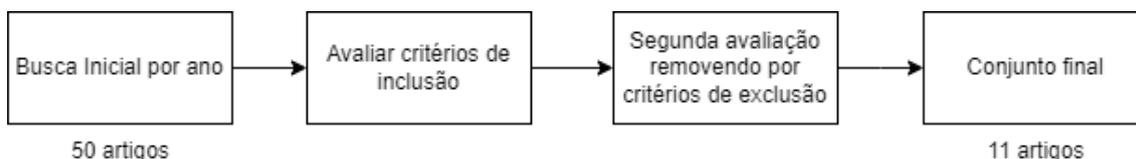
- Não estejam diretamente relacionados ao tema da inteligência artificial gerativa
- Não duplicados
- Não fizeram algum tipo de avaliação quantitativa ou qualitativa dos resultados obtidos
- Tem foco na geração de outros artefatos para jogos, como áudio ou imagem

No processo utilizado foram enumerados os 10 primeiros artigos publicados em cada ano que contemplem os critérios de inclusão, totalizando 50 artigos que posteriormente foram filtrados segundo os critérios de exclusão, como ilustrado na figura 4. Após todo o processo restaram 11 artigos:

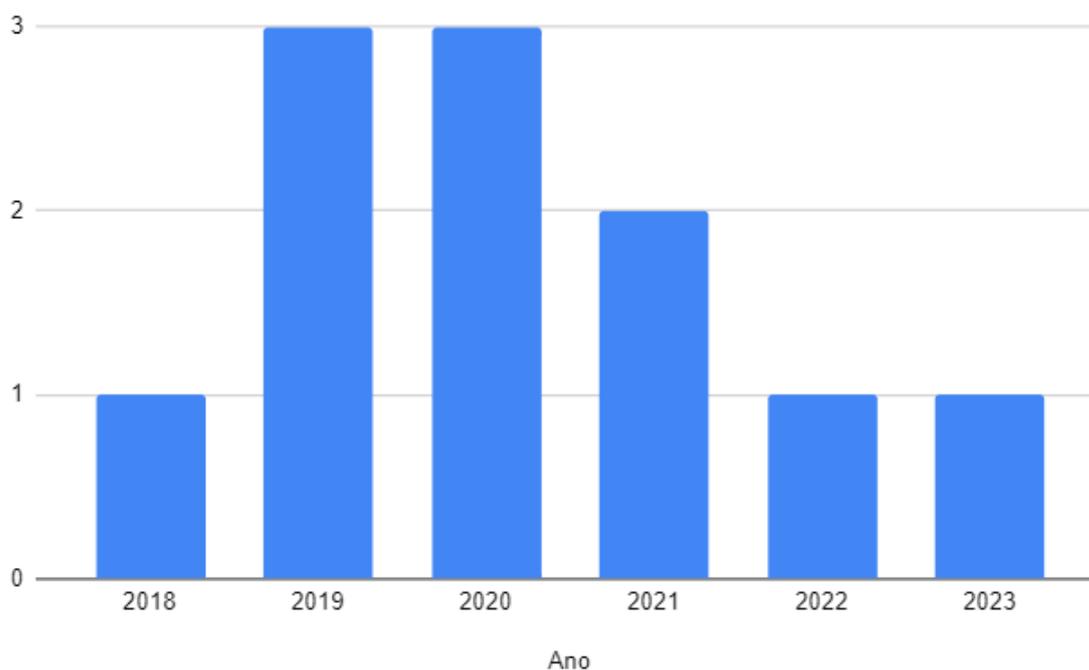
- *Evolving mario levels in the latent space of a deep convolutional generative adversarial network*
- *A dynamic balanced level generator for video games based on deep convolutional generative adversarial networks*
- *Generative Adversarial Network Rooms in Generative Graph Grammar Dungeons for The Legend of Zelda*
- *Interactive evolution and exploration within latent level-design space of generative adversarial networks*
- *Evolving Levels for General Games Using Deep Convolutional Generative Adversarial Networks*
- *Compressing and Comparing the Generative Spaces of Procedural Content Generators*
- *Study on Stages Evaluation for 2D Action Game Generated by DC-GAN Based on Artificial Intelligence*
- *TOAD-GAN: Coherent Style Level Generation from a Single Example*
- *Searching the Latent Space of a Generative Adversarial Network to Generate DOOM Levels*
- *Using Multiple Generative Adversarial Networks to Build*
- *Better-Connected Levels for Mega Man*
- *Procedural Content Generation of Super Mario Levels Considering Natural Connection*

Após filtrar o conjunto de artigos analisaremos os principais modelos utilizados, a qualidade dos resultados obtidos, as técnicas utilizadas, além das limitações e possíveis próximos passos a serem tomados.

**Figura 4. Metodologia para construir conjunto de artigos.**



**Figura 5. Distribuição dos anos de publicação dos estudos analisados.**



## 4. Resultados

### 4.1. Sobre os artigos

Seguindo a abordagem proposta, um conjunto de 11 artigos que contemplam o tema foram analisados. Destes a maior parte de 2019 e 2020 como ilustrado na figura 5.

No que diz respeito aos jogos que foram trabalhados os dois jogos mais abordados foram Super Mario Bros seguido por Zelda, além disso um dos estudos se baseia na geração de níveis para um jogo fictício, a distribuição completa dos jogos utilizados nos estudos pode ser visto na figura 6.

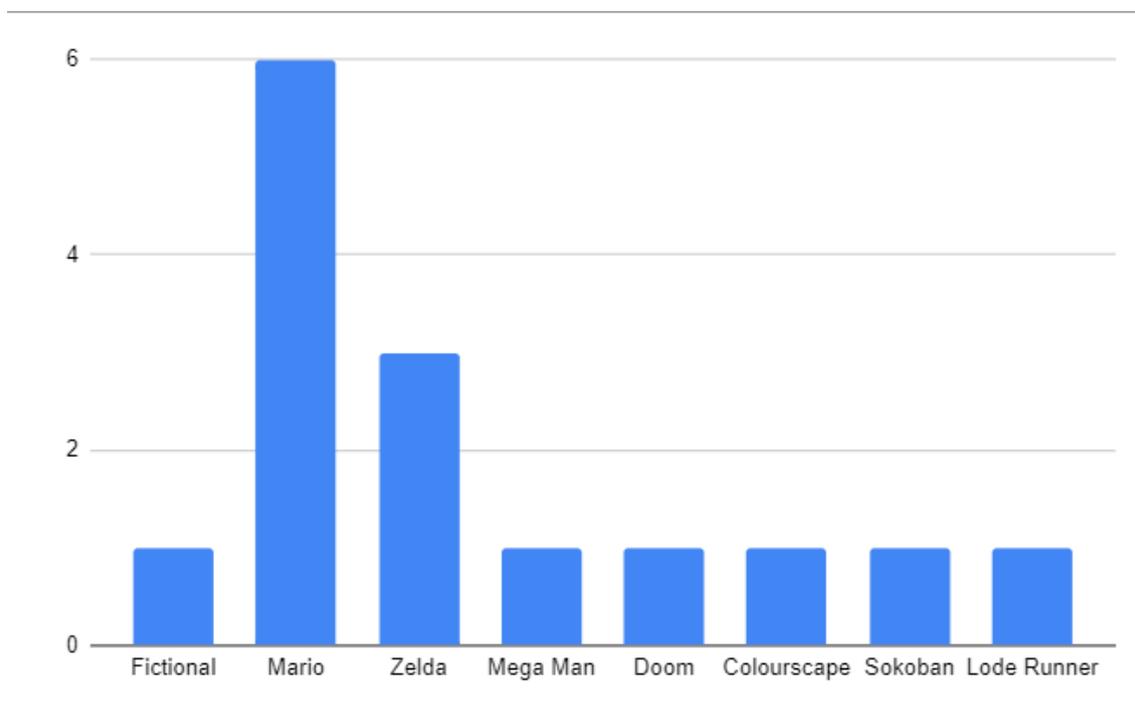
Em se tratando dos modelos GAN o mais utilizado foi o DCGAN como ilustrado na figura 7.

### 4.2. Espaços explorados

Existe um vasto espaço de possíveis jogos que podem ser abordados para esse tipo de geração procedural de níveis, entretanto nos artigos analisados os autores em sua maioria se restringiram em que os níveis podem ser representados por modelos 2D.

Além disso, é válido destacar que os estudos se utilizam de duas principais abordagens, o uso de modelos baseados em imagens e o uso de modelos baseados em textos.

**Figura 6. Distribuição dos jogos abordados nos estudos analisados.**



Enquanto alguns modelos converteram os níveis do conjunto de exemplos reais em um conjunto de tokens textuais, outros artigos utilizavam diretamente imagens, processo que se mostrou mais genérico que o anterior. Além disso é válido dar destaque para as técnicas utilizadas no estudo de 2019 para simplificar a geração de níveis que mais se aproximam do 3D do jogo Doom [Giacomello et al. 2019].

### **4.3. Desafios**

Na geração de conteúdo procedural(PCG) algumas métricas podem ser utilizadas como critérios de qualidade, a exemplo de: jogabilidade, credibilidade, dificuldade, entretenimento e naturalidade [Nam et al. 2023].

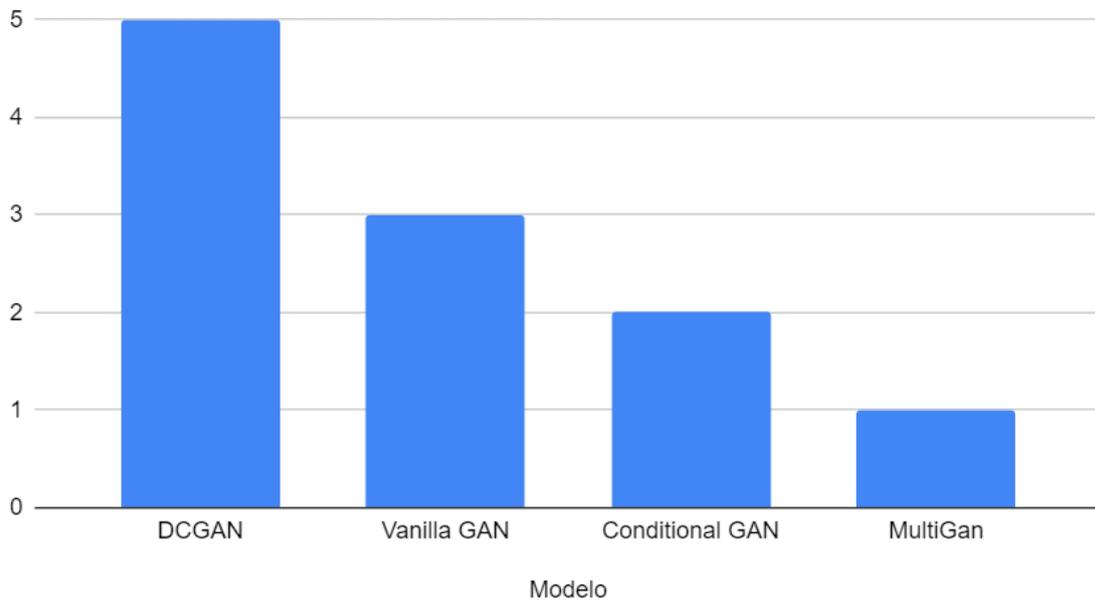
Um bom modelo de geração procedural deve garantir que os níveis gerados possuam uma estrutura lógica, jogabilidade equilibrada e fluidez, esses são os maiores desafios a serem superados pelos modelos utilizados na geração de níveis.

### **4.4. Abordagens**

#### **4.4.1. Geração com texto**

Alguns dos estudos optaram por utilizar modelos que traduzem os níveis em tokens textuais [Volz et al. 2018, Gutierrez and Schrum 2020, Schrum et al. 2020, Capps and Schrum 2021, Nam et al. 2023]. Neste processo os mapas são divididos em unidades chamadas de *tiles* que são então associadas a uma palavra chave, como observado na figura 8, onde por exemplo o *tile* de piso é associado ao valor 0. Utilizando o processo de aprendizagem adversarial os níveis são traduzidos e utilizados no treinamento dos modelos GAN. Após o treinamento o gerador é capaz de gerar níveis que serão traduzidos de volta para o seu formato original.

**Figura 7. Distribuição dos modelos abordados nos estudos analisados.**



#### **4.5. Baseado em imagens**

Outros estudos optaram por utilizar imagens diretamente nos modelos GAN [Rajabi et al. 2021, Irfan et al. 2019, Withington and Tokarchuk 2022, Nagahiro et al. 2019, Awiszus et al. 2020, Giacomello et al. 2019]. Uma das principais vantagens dessa abordagem é ser uma abordagem mais genérica permitindo utilizar o mesmo processo para múltiplos jogos [Nagahiro et al. 2019].

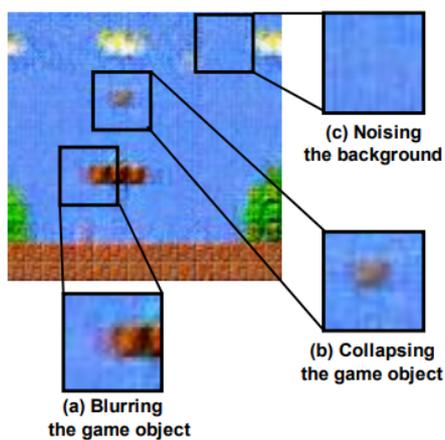
O uso direto das imagens possui alguns problemas como objetos borrados e ruídos no plano de fundo como observado na figura 9 . Para solucionar esse problema a abordagem eficiente parece ser um pré processamento das imagens de forma similar à abordagem anterior. Um pré-processamento das imagens é feito para que possam ser utilizadas no treinamento dos modelos. Nesse caso a imagem é simplificada em uma imagem de menor resolução onde cada pixel pode ser associado a um *tile* através de seu valor RGB como indicado na figura 10. O resultado do processo mencionado pode ser observado na figura 11 .

Figura 8. Exemplo de valores utilizados para representar valores de tiles em Mario.

Tile type	Symbol	Identity	Visualization
Solid/Ground	X	0	
Breakable	S	1	
Empty (passable)	-	2	
Full question block	?	3	
Empty question block	Q	4	
Enemy	E	5	
Top-left pipe	<	6	
Top-right pipe	>	7	
Left pipe	[	8	
Right pipe	]	9	

Fonte: [Volz et al. 2018]

Figura 9. Problemas apresentados pelo uso direto de DCGAN sem processamento na imagem dos níveis.



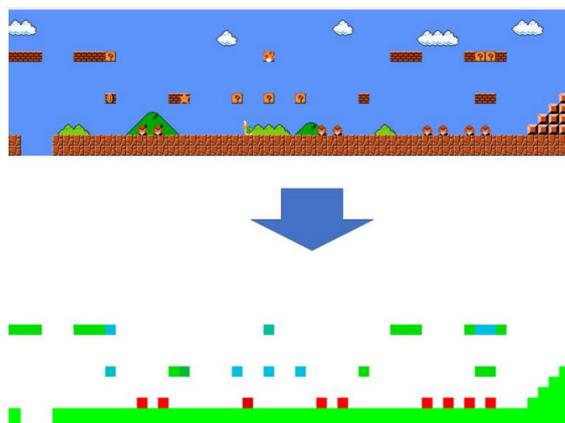
Fonte: [Nagahiro et al. 2019].

Figura 10. Tabela associando uma faixa de valores RGB a elementos específicos do nível.

Enemy (R)			Block (G)			Item (B)		
dot	color	pixel	dot	color	pixel	dot	color	pixel
		255			255			255
		213			204			170
		170			153			85
		128			102			
		85			51			
	43							

Fonte: [Nagahiro et al. 2019].

**Figura 11. Pré-processamento de mapa de "Super Mario Brothers" para treinamento com DCGAN.**



Fonte: [Nagahiro et al. 2019].

#### 4.6. DOOM

É válido abordar o caso do estudo de geração de níveis de DOOM [Giacomello et al. 2019], com o uso de algumas técnicas o autor foi capaz de gerar níveis que se afastam da área de jogos 2D abordados pelos outros estudos.

Os níveis do jogo são armazenados em um conjunto de arquivos WAD contendo as informações sobre cada nível. O autor traduz os arquivos WAD em 6 imagens que podem ser utilizadas para reconstruir os níveis. Além disso um conjunto de features foram extraídas das imagens e arquivos WAD para auxiliar o discriminador a classificar os níveis reais.

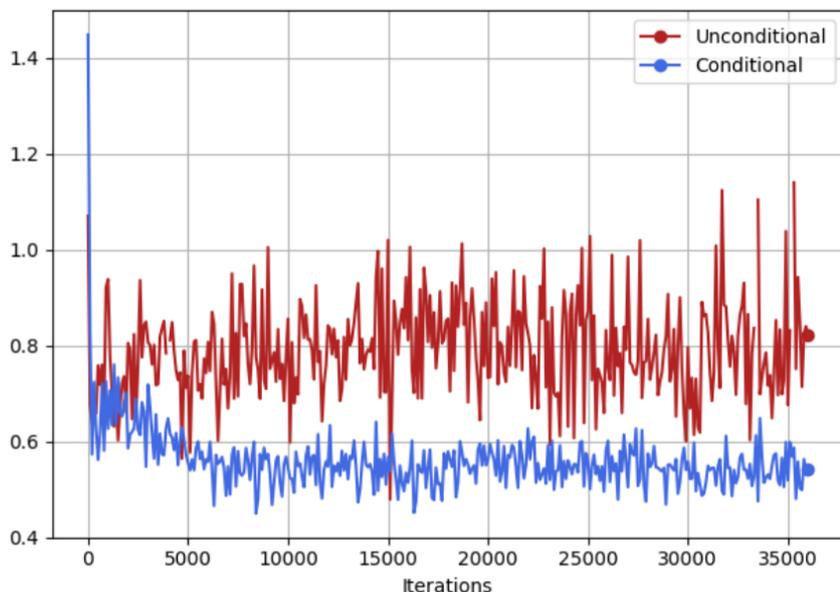
O autor utilizou um modelo GAN condicional e um incondicional e realizou o treinamento utilizando um conjunto de 1000 níveis de DOOM. Ambos os modelos demonstraram resultados promissores mantendo diversas características dos níveis originais. Exemplos de níveis gerados por GAN neste estudo podem ser observados na figura 14.

Dentre as 6 imagens utilizadas pelo autor, duas são utilizadas para comparar os modelos: o mapa do piso e o mapa de paredes de cada nível. O autor compara os dois modelos utilizando uma métrica de Corner Error (CE), calculando a diferença entre o número de cantos em um mapa gerado por um modelo GAN e um mapa feito por um humano, a relação do CE no treinamento dos modelos podem ser observados para os mapas de paredes e pisos respectivamente nas figuras 12 e 13.

#### 4.7. MultiGan

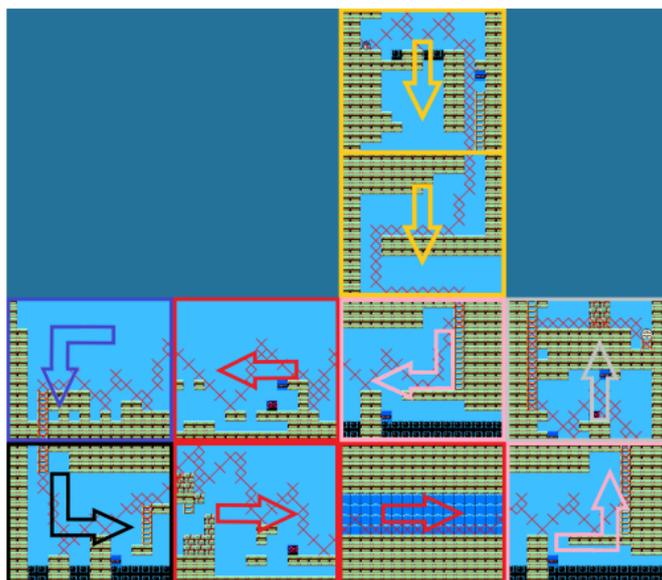
Outra técnica abordada em um único estudo foi o uso de múltiplos modelos GAN, em estudo de 2021 [Capps and Schrum 2021] o autor se utiliza de uma arquitetura de MultiGan para produzir níveis do jogo Mega Man. O autor se utiliza desse tipo de modelo com o objetivo de criar GANs específicos para cada tipo de segmento de um dado nível como ilustrado na figura 15. Cada GAN é especialista em produzir um tipo específico de segmento, como os trechos em vermelho onde o jogador se desloca pelo mapa horizontalmente.

Figura 12. CE observado nos mapas de paredes de Doom ao longo do treinamento dos modelos condicional e não condicional, onde 0 seria o mesmo número de cantos que um mapa gerado por humano.



Fonte: [Giacomello et al. 2019].

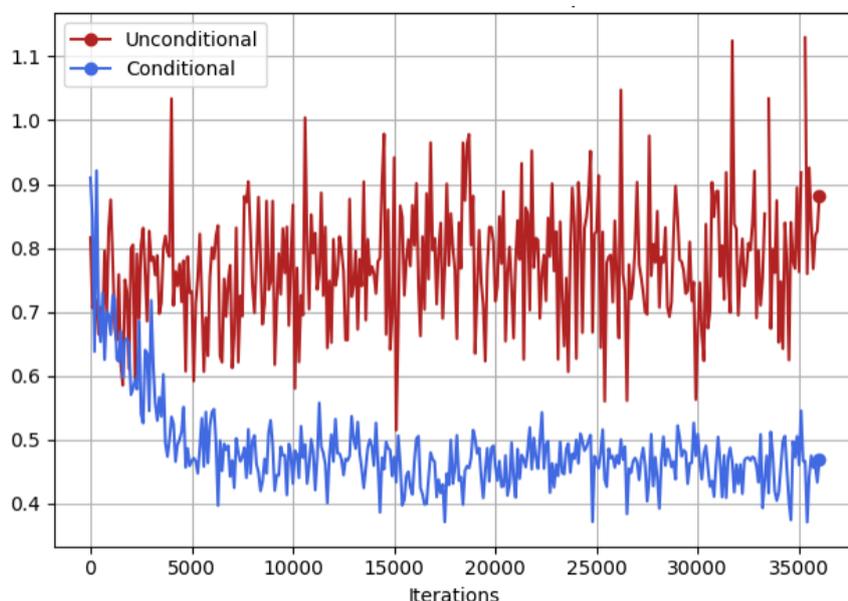
Figura 15. Nível de Mega Man gerado com o uso de uma arquitetura do tipo MultiGan onde modelos GAN especialistas geram trechos de formatos específicos.



Fonte: [Capps and Schrum 2021].

Em estudos comparativos entre MultiGan e um único GAN os resultados apontam que nesse jogo em específico o uso do MultiGan apresenta melhores resultados nos quesitos de dificuldade, diversão e similaridade com níveis produzidos por humanos como

Figura 13. CE observado nos mapas de paredes de Doom ao longo do treinamento dos modelos condicional e não condicional, onde 0 seria o mesmo número de cantos que um mapa gerado por humano.



Fonte: [Giacomello et al. 2019].

ilustrado na figura 16.

Figura 16. Resultado de teste com 20 usuários comparando um único GAN com modelo MultiGan.

Question	Type	OneGAN	MultiGAN	$p$
Completed?	y/n	14	17	N/A
Platform-Gun Required?	y/n	11	8	N/A
Created by a Human?	y/n	4	13	N/A
Harder?	e/o	14	6	0.05766
More Fun?	e/o	2	18	<b>0.0002012</b>
More Human-Like?	e/o	5	15	<b>0.02069</b>
More Interesting?	e/o	7	13	0.1316

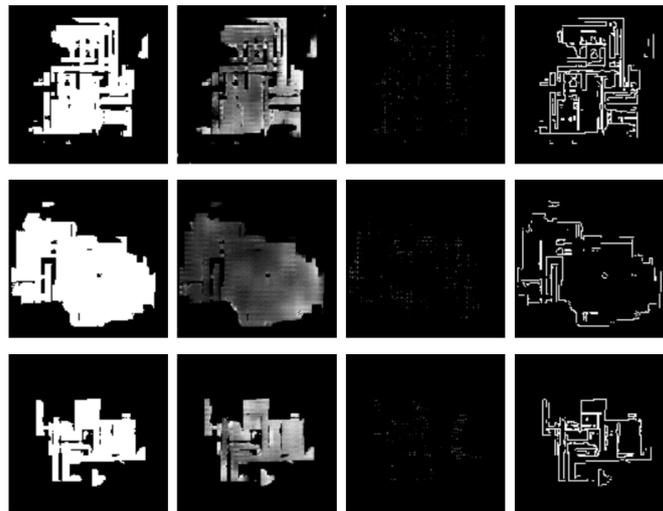
Fonte: [Capps and Schrum 2021].

#### 4.8. Navegando pelo espaço de níveis

Dentre os estudos analisados alguns deles procuraram produzir uma ferramentas que que auxiliam o usuário final a navegar pelo espaço de níveis gerados, permitindo que um usuário final possa explorar o espaço latente modificando diretamente os valores dos vetores latentes que serão mapeados para um nível, apesar da navegação pouco conveniente este tipo de ferramenta pode proporcionar a criação de níveis por um usuário final sem necessariamente ter conhecimentos prévios de modelos de GAN.

Em um estudo de 2020 [Schrum et al. 2020] o autor desenvolveu 2 tipos de abordagens de ferramentas de design de níveis, sendo uma delas baseada em algoritmo

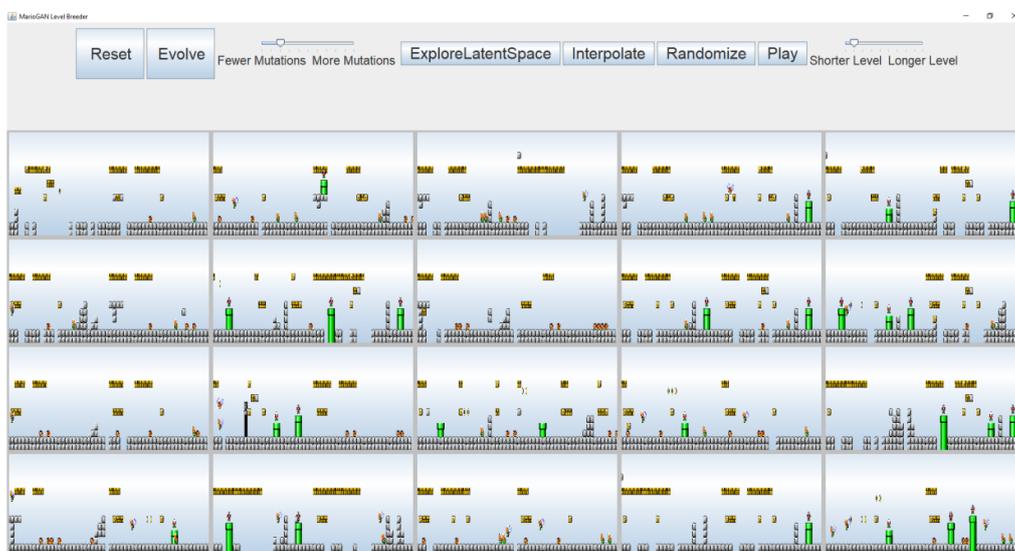
Figura 14. Níveis gerados utilizando modelo GAN incondicional.



Fonte: [Giacomello et al. 2019].

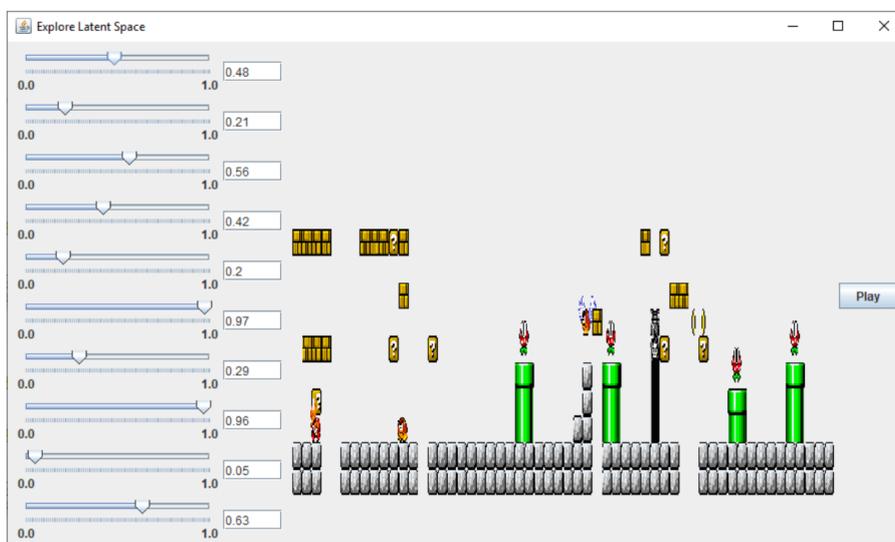
genético, onde o usuário seleciona manualmente quais níveis são 'melhores' e os vetores dos níveis selecionados serão utilizados para gerar a próxima geração de níveis, a interface pode ser observada na figura 17. Outra abordagem proposta no mesmo estudo foi o uso inputs de usuário para afetar diretamente os valores utilizados como input do modelo GAN como demonstrado na figura 18.

Figura 17. Ferramenta de design de níveis utilizando algoritmo genético associado à geração com GAN, sendo utilizada para gerar níveis de Super Mario Bros.



Fonte: [Schrum et al. 2020].

**Figura 18. Ferramenta de design de níveis utilizando exploração direta do espaço latente, sendo utilizada para gerar níveis de Super Mario Bros.**

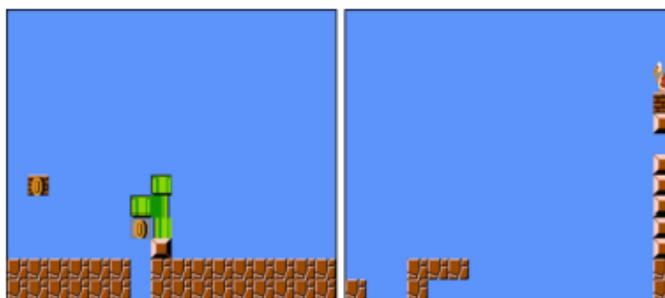


Fonte: [Schrum et al. 2020].

#### 4.9. Coesão

Nos níveis gerados por GAN é possível que existam incoerências que diminuem a naturalidade e qualidade dos níveis gerados, alguns dos autores se utilizaram de técnicas para tentar gerar níveis mais coesos. Alguns resultados incoerentes gerados por GAN podem ser observados na figura 19, na imagem da esquerda os *tiles* dos canos não estão dispostos da forma correta, na imagem da direita há uma região sem piso que torna impossível para completar o nível.

**Figura 19. Resultados de níveis incoerentes gerados por GAN.**

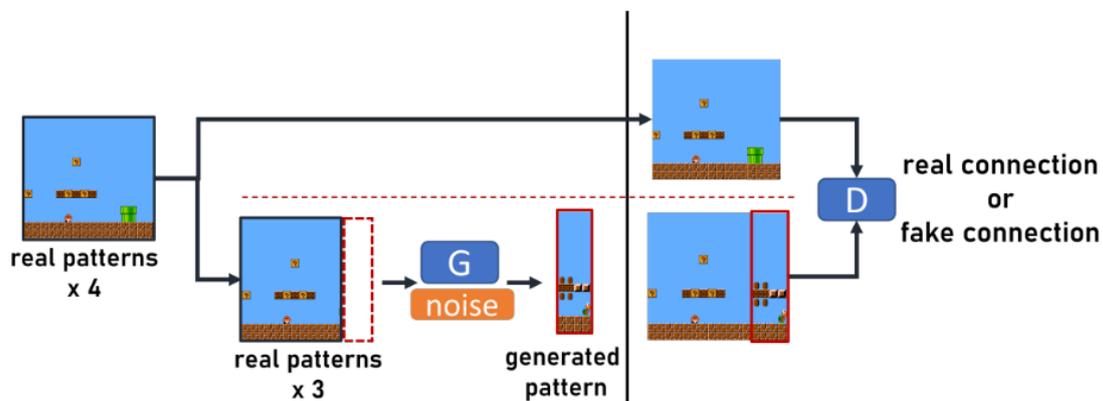


Fonte: [Nagahiro et al. 2019].

Uma das possíveis abordagens utilizadas para atacar esse problema em específico é o uso de CGAN, utilizando o CGAN onde o GAN gera os níveis coluna a coluna [Nam et al. 2023]. Nessa abordagem as colunas anteriores são utilizadas como entradas no treinamento do gerador e discriminador, condicionando o gerador a criar a coluna atual de forma condizente com as colunas anteriores, criando uma coesão nos níveis gerados. O processo de treinamento é ilustrado na figura 20, onde um conjunto de 3 colunas é usado como entrada para gerar a coluna seguinte. Além disso, esse tipo de

ferramenta pode proporcionar um processo de co-criação onde o usuário pode selecionar colunas específicas para serem geradas pelo modelo GAN.

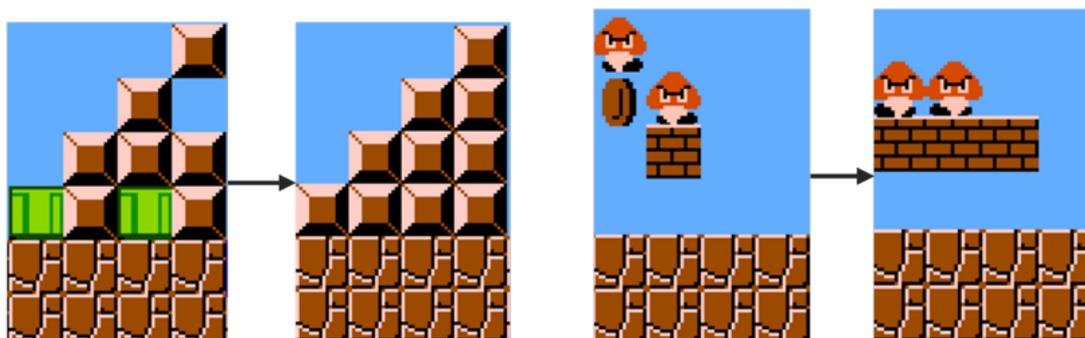
**Figura 20. Treinamento de CGAN com base nas colunas anteriores.**



Fonte: [Nam et al. 2023].

Outra técnica que foi utilizada com sucesso para aumentar a coesão nos níveis gerados é uso de padrões conhecidos dentro dos níveis, com essa técnica os padrões gerados pelo gerador são convertidos em vetores normalizados e comparados com os padrões existentes nos dados originais para então serem substituídos pelo padrão mais próximo existente nos níveis originais [Nam et al. 2023]. Exemplos de padrões próximos para substituição podem ser observados na figura 21. Apesar de diminuir a quantidade de padrões originais e criativos, essa técnica é capaz de gerar mapas consideravelmente mais próximos aos dados originais.

**Figura 21. Padrões incoerentes gerados por GAN sendo substituídos por padrões originais mais próximos.**



Fonte: [Nam et al. 2023]

#### 4.10. Fontes de dados

Para a obtenção de dados para o treinamento dos modelos, diversas abordagens foram adotadas:

- Usar um grande conjunto de níveis disponíveis [Giacomello et al. 2019]

- Dividir cada nível do jogo em múltiplas telas [Volz et al. 2018]
- Dividir os níveis em múltiplas colunas [Nam et al. 2023]
- Usar outros modelos de IA para gerar um conjunto de níveis e filtrar os 'bons' [Rajabi et al. 2021]

Com o uso das técnicas mencionadas os autores foram capazes de obter dados contornando quando necessário a falta ou ausência de dados necessários para o treinamento dos modelos.

#### 4.11. Da qualidade dos resultados

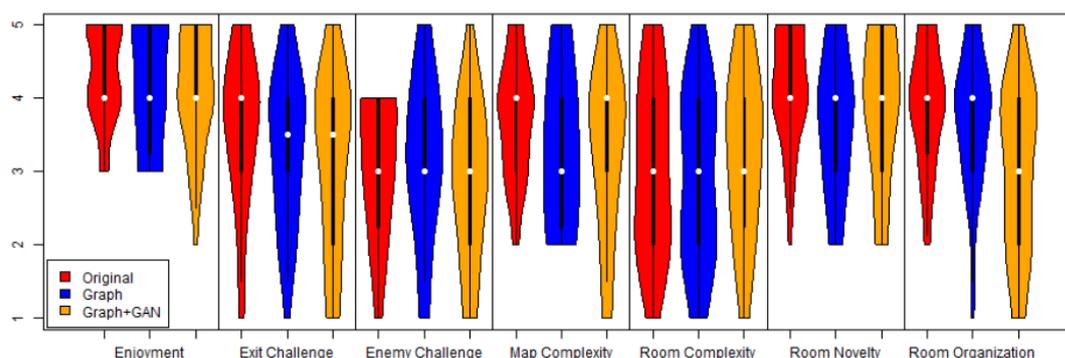
Dentre os estudos observados nenhum apresentou resultado com qualidade suficiente para ser utilizado sem algum tipo de supervisão humana, apesar disso algumas métricas foram utilizadas para avaliar a qualidade dos resultados obtidos, algumas das quais são válidas de menção.

##### 4.11.1. Avaliação com usuários

Dois dos estudos verificados validaram os resultados com usuários. Dentre os experimentos feitos, em um deles, de 2020, o autor faz uma avaliação com 30 pessoas. Três tipos de níveis são comparados: níveis originais do jogo, níveis gerados utilizando reescrita de grafos e níveis gerados utilizando um modelo GAN associado novamente à reescrita de grafos.

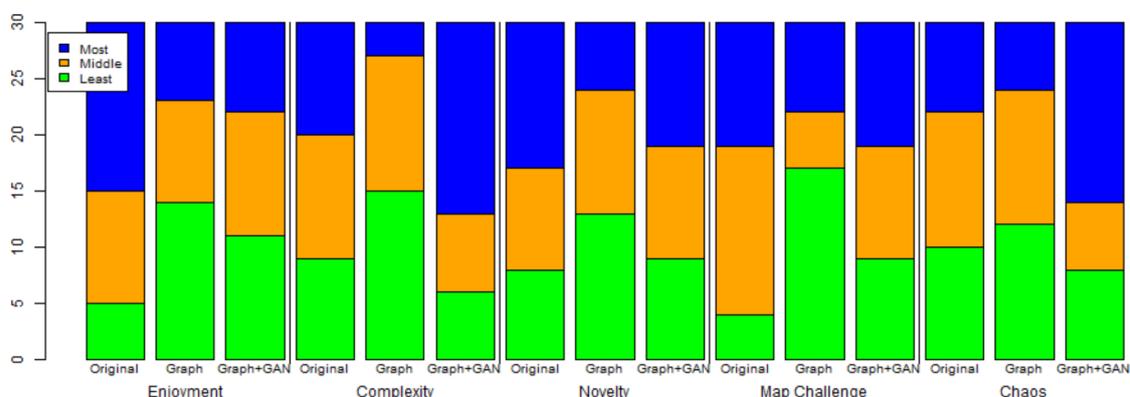
Cada participante jogou um de cada nível e os avaliou com uma nota de 1 a 5 nas categorias 'Satisfação', 'Desafio para encontrar saída', 'Desafio dos inimigos', 'Complexidade do mapa', 'Complexidade das salas', 'Novidade nas salas' e 'Organização nas salas', além disso os participantes também ranquearam cada nível nas categorias 'Satisfação', 'Complexidade', 'Novidade', 'Desafio do mapa' e 'Caos', com os resultados da avaliação dos usuários nas figuras 22 e 23 [Gutierrez and Schrum 2020].

**Figura 22. Resultado de avaliações com usuários de geração de níveis de Zelda utilizando os modelos Graph + GAN.**



Fonte: [Gutierrez and Schrum 2020].

**Figura 23. Resultado de avaliações com usuários de geração de níveis de Zelda utilizando os modelos Graph + GAN.**



Fonte: [Gutierrez and Schrum 2020].

Em outro estudo feito em 2020 o autor desenvolveu uma ferramenta de design de níveis com o uso de modelos GAN [Schrum et al. 2020], para avaliar as ferramentas o autor fez um teste com 22 usuários. O autor desenvolveu questionou os usuários com relação à preferência no uso das ferramentas, se os usuários preferiam produzir o design dos níveis navegando pelo espaço dos níveis ou utilizando um algoritmo de evolução, e questionou também sobre a combinação de ambos os métodos, os resultados podem ser observados na figura 24.

**Figura 24. Resultado de pesquisa com usuários em relação à preferência sobre ferramenta de design de níveis.**

	Mario	Zelda	Total
Prefers Evolution	3 (33.33%)	5 (41.67%)	8 (38.10%)
Prefers Exploration	6 (66.67%)	7 (58.33%)	13 (61.90%)
Combination Worse	1 (10.00%)	0 (0.00%)	1 (4.55%)
Combination Equal	3 (30.00%)	4 (33.33%)	7 (31.82%)
Combination Better	6 (60.00%)	8 (66.67%)	14 (63.64%)

Fonte: [Schrum et al. 2020].

#### 4.11.2. Busca de características específicas

Dentre os estudos feitos um deles buscou avaliar a qualidade de níveis construídos com algumas características específicas, o autor procurou gerar um nível de Super Mario Bros dividindo o mapa dividido em 5 seções sendo as duas primeiras com 100% de piso e as 3 seguintes 70% de cobertura, além disso o autor maximiza o número de inimigos gerados nas seções 4 e 5 [Volz et al. 2018], o resultado obtido pode ser observado na figura 25.

**Figura 25. Nível de Super Mario Bros gerado em 5 seções sendo as 2 primeiras 100% cobertas por terreno, as 3 últimas 70% cobertas por terreno e maximizando o número de inimigos gerados nas seções 4 e 5.**



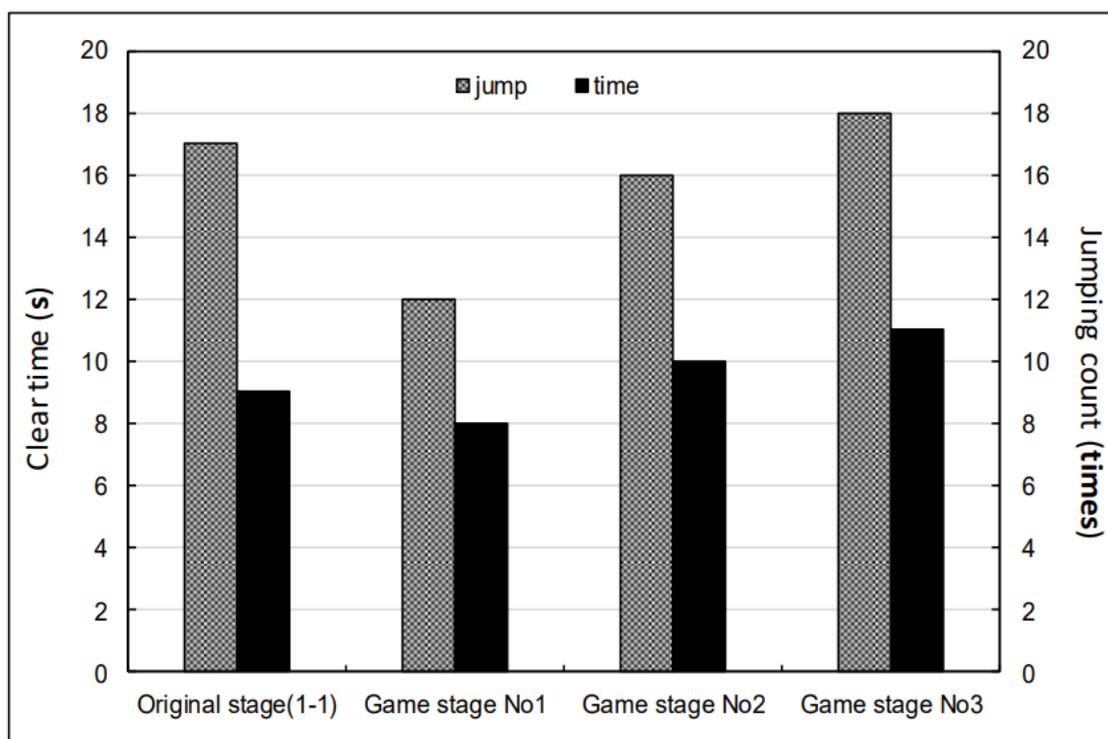
Fonte: [Volz et al. 2018]

#### 4.12. Solução com Agentes

O método de validação que foi mais utilizado nos estudos analisados foi o teste com agentes treinados por aprendizagem por reforço ou com o uso do algoritmo A\* [Rajabi et al. 2021, Irfan et al. 2019, Nagahiro et al. 2019, Awiszus et al. 2020, Capps and Schrum 2021].

Dentre as técnicas válidas de menção um dos autores procurou comparar o número de pulos necessários para completar os níveis considerando a solução encontrada com o algoritmo A\* nos níveis originais e nos níveis gerados [Nagahiro et al. 2019], os resultados obtidos podem ser observados na figura 26.

**Figura 26. Relação de número de pulos e tempo para completar níveis de Super Mario Bros originais e gerados por modelo GAN.**



Fonte: [Nagahiro et al. 2019].

Outro dos métodos mencionados foi o uso de agentes, o autor utilizou o framework GVG-AI para testar a solvibilidade dos níveis gerados dos jogos freeway, zelda e colorscape utilizando DCGAN [Irfan et al. 2019].

**Figura 27. Validação de níveis dos jogos Freeway, Zelda e Colourscape gerados com o modelo DCGAN validados por um agente do GVG-AI.**

Game	Generated Levels	Playable	Unplayable	Win Percentage
Freeway	25	22	3	87%
Zelda	25	21	4	81%
Colourscape	25	21	4	81%

Fonte: [Irfan et al. 2019].

#### **4.13. Discussão**

Em suma, a aplicação da Inteligência Artificial gerativa(IAG) na criação de níveis em jogos digitais representa uma evolução notável na indústria dos videogames. Este artigo explorou os avanços dessa tecnologia, destacando como ela pode aprimorar o trabalho do game designer, reduzindo o tempo de desenvolvimento e estimular a criatividade dos desenvolvedores. A IAG oferece a capacidade de criar níveis complexos de forma automatizada, economizando recursos preciosos de desenvolvimento.

Com o avanço de modelos GAN a produção de ferramentas que permitam a navegação pelo espaço de níveis produzidos por IA gerativa pode ser considerada uma realidade. A criação de níveis equivalentes aos produzidos por humanos entretanto ainda não é possível, apesar disso as ferramentas existentes já apresentam resultados coesos e originais, que podem auxiliar na processo criativo do design de layouts de níveis.

No entanto, é fundamental reconhecer os desafios que essa abordagem enfrenta, como a necessidade de treinar algoritmos adequados e a preocupação com a criatividade humana perdida no processo. Portanto, a harmonização entre IA e criatividade humana é essencial para aproveitar ao máximo o potencial da IAG na criação de níveis de jogos.

À medida que a tecnologia continua a avançar, é possível que a IA gerativa ganhe espaço na indústria de jogos digitais, expandindo os limites da criatividade e melhorando o processo de desenvolvimento. Com uma abordagem equilibrada e a combinação das habilidades únicas da inteligência artificial e da mente humana, podemos esperar grandes avanços nos jogos digitais.

### **5. Conclusões e Trabalhos Futuros**

A análise dos estudos indica que os avanços nesse campo são mais restritos aos jogos 2D, apesar do campo do uso de GANs já apresentar resultados na geração de artefatos 3D, tendo somente um dos estudos abordado um jogo não totalmente bidimensional.

As observações sugerem que o uso de modelos GAN na geração de níveis já fez grandes avanços, sendo possível treinar e utilizar modelos com quantidades reduzidas de dados de treinamento se utilizando diversas das técnicas apresentadas para contornar o problema da falta ou ausência de dados. O desafio de encontrar dados que é ainda maior para jogos que ainda estão em desenvolvimento, pode ser superado como demonstrado no estudo de 2021 onde o autor sugere o uso de técnicas para gerar níveis para um jogo fictício.

Seria ideal ter um grande conjunto de dados disponíveis, mas este não é um cenário realista para a maior parte dos casos. Contudo mesmo com bases de dados re-

lativamente pequenas os estudos apresentaram bons resultados. No estudo de 2019 onde o autor utiliza cerca de 1000 níveis produzidos a mão de Doom [Giacomello et al. 2019]. O autor se aproveita de um grande número de níveis disponíveis e vai além do que a maior parte dos estudos foi capaz de obter, produzindo conteúdo que se aproxima mais do 3D. Entretanto é mais razoável considerar a existência de pouco ou nenhum dado à disposição dos desenvolvedores de jogos como foi o caso da maior parte dos estudos.

Embora os resultados dos estudos observados indicam que o resultado gerado pelos modelos GAN ainda não são equivalentes aos feitos por humanos e ainda contenham algumas incongruências, no estado atual é possível que os modelos GAN sejam utilizados como ferramentas para auxiliar game designers a idealizar e planejar níveis, contudo certamente ainda não é possível substituir o agente humano gerando níveis que contemplem a complexidade, diversão, balanceamento e dificuldade desejada em um nível artefato finalizado. Apesar disso muitas técnicas foram utilizadas com sucesso para manter se aproximar dessas características e para melhorar a coesão dos artefatos gerados.

No que diz respeito à qualidade dos modelos apresentados é difícil comparar os modelos de forma objetiva considerando que os estudos seguem diferentes métodos de validação. Entretanto um dos estudos chega a comparar diretamente um modelo GAN condicional com um não condicional obtendo resultados favoráveis para o modelo condicional[Giacomello et al. 2019]. Além disso foi demonstrado que algumas técnicas podem ser utilizadas para usar o input de modelos condicionais para melhorar a coesão de um nível gerado por GAN, mais uma vez favorável ao uso desse tipo de modelo.[Nam et al. 2023]

Outro modelo abordado foi o MultiGan, que apresenta melhoras em relação a um único GAN, mas seu uso só faz sentido em jogos onde é possível segmentar a geração de níveis de forma similar ao proposto pelo autor na geração de níveis de Mega Man. Nenhum estudo comparou o uso de MultiGan com GANs condicionais.

Em relação aos a métricas mais abstratas observadas níveis gerados com IA gerativa um estudo de 2020 indica que os níveis gerados por IA gerativa apesar de ainda é inferior em diversos aspectos como pode ser observado na figura 22, onde fica evidenciado como os modelos GAN são inferiores a níveis feitos a mão em aspectos como organização e desafio.

Manter a coesão se demonstrou um dos maiores desafios enfrentados na geração de níveis com IA gerativa, nos resultados obtidos é comum observar incongruências nos mapas gerados, algumas das quais tornam o mapa não só incoerente mas também impedem que os jogadores possam completar o nível gerado. Algumas técnicas melhoram a coesão dos níveis gerados como o uso dos modelos condicionais para gerar níveis de forma linear se baseando em colunas anteriores já geradas para gerar as colunas seguintes mantendo a coesão. Além disso o uso de técnicas de substituição de partes de níveis já gerados por partes similares no corpo de níveis originais, apesar de diminuir a originalidade dos níveis gerados pode proporcionar resultados mais próximos aos níveis originais.

Um dos possíveis próximos passos para a área é o uso de modelos GAN para gerar níveis em jogos 3D. A escassez de trabalhos nessa área pode se dar pela dificuldade de se trabalhar com artefatos de maior dimensionalidade, exigindo modelos mais complexos, maior poder computacional e também da dificuldade de obter dados de treinamento.

Alguns dos artigos já começam a pavimentar possíveis alternativas para avançar nessa área.

## Referências

- Awiszus, M., Schubert, F., and Rosenhahn, B. (2020). Toad-gan: Coherent style level generation from a single example.
- Capps, B. and Schrum, J. (2021). Using multiple generative adversarial networks to build better-connected levels for mega man.
- Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., and Bharath, A. A. (2018). Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65.
- Gao, L., Chen, D., Zhao, Z., Shao, J., and Shen, H. T. (2021). Lightweight dynamic conditional gan with pyramid attention for text-to-image synthesis. *Pattern Recognition*, 110:107384.
- Giacomello, E., Lanzi, P. L., and Loiacono, D. (2019). Searching the latent space of a generative adversarial network to generate doom levels. In *2019 IEEE Conference on Games (CoG)*, pages 1–8.
- Gutierrez, J. and Schrum, J. (2020). Generative adversarial network rooms in generative graph grammar dungeons for the legend of zelda. *CoRR*, abs/2001.05065.
- Irfan, A., Zafar, A., and Hassan, S. (2019). Evolving levels for general games using deep convolutional generative adversarial networks. In *2019 11th Computer Science and Electronic Engineering (CEECE)*, pages 96–101.
- Nagahiro, K., Ooi, S., and Sano, M. (2019). Study on stage generation for 2d action game based on dc-gan. pages 79–85.
- Nam, S., Hsueh, C.-H., and Ikeda, K. (2023). Procedural content generation of super mario levels considering natural connection. In *2023 20th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 291–296.
- Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks.
- Rajabi, M., Ashtiani, M., Minaei Bidgoli, B., and Davoodi, O. (2021). A dynamic balanced level generator for video games based on deep convolutional generative adversarial networks. *Scientia Iranica*, 28(Special issue on collective behavior of nonlinear dynamical networks):1497–1514.
- Schrum, J., Gutierrez, J., Volz, V., Liu, J., Lucas, S., and Risi, S. (2020). Interactive evolution and exploration within latent level-design space of generative adversarial networks. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*, GECCO '20, page 148–156, New York, NY, USA. Association for Computing Machinery.
- Volz, V., Schrum, J., Liu, J., Lucas, S. M., Smith, A., and Risi, S. (2018). Evolving mario levels in the latent space of a deep convolutional generative adversarial network. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '18, page 221–228, New York, NY, USA. Association for Computing Machinery.

Withington, O. and Tokarchuk, L. (2022). Compressing and comparing the generative spaces of procedural content generators. In *2022 IEEE Conference on Games (CoG)*. IEEE.