

Geração de Loteamentos Residenciais com Deep Learning baseada em Redes Adversárias Generativas

Milena Lucas dos Santos
Centro de Tecnologias Aplicadas
Itaipu Parquetec
Foz do Iguaçu, Brasil
milena.santos@itaipuparquetec.org.br

Rafhael Pereira Zrenner
Centro de Tecnologias Aplicadas
Itaipu Parquetec
Foz do Iguaçu, Brasil
rafael.zrenner@itaipuparquetec.org.br

Fabiana Frata Furlan Peres
EACH/USP
CECE / UNIOESTE
Foz do Iguaçu, Brazil
fabiana.peres@unioeste.br
<https://orcid.org/0000-0002-3937-3432>

Érick Oliveira Rodrigues
Pós graduação em Engenharia de Produção e Sistemas (PPGEPS)
Universidade Tecnológica Federal do Paraná (UTFPR)
Pato Branco, Brasil
erickr@lncc.br

Resumo—This work presents an exploratory study on the use of Pix2Pix generative networks to support the Architecture and Engineering team at Itaipu Parquetec in the automatic generation of preliminary layouts for residential subdivisions. The proposal seeks to assist the early stages of urban planning by creating layouts based on predefined parameters such as block, lot, and land dimensions. To this end, a dataset of 700 samples was developed, built from images generated in AutoCAD combined with real urban grid maps of Foz do Iguaçu, Brazil. The methodology involved data preprocessing, road network generation, conditional Generative Adversarial Network (GANs) training, and hyperparameter tuning. Quantitative evaluation using the Fréchet Inception Distance (FID) yielded values of 40.61 for simple scenarios and 73.09 after including the road network, confirming the increased complexity of the task. Nevertheless, the approach can be used as a support tool in the initial stages of urban planning. Future work includes expanding the dataset to cover different urban typologies, making it available as a benchmark for the field, and investigating the use of Diffusion Models as an alternative to improve the geometric fidelity of the generated layouts.

Keywords—Generative Adversarial Networks; Pix2Pix; Generative Design; Residential Layout.

Resumo—Este trabalho apresenta um estudo exploratório do uso de redes generativas Pix2Pix para apoiar a equipe de Arquitetura e Engenharia do Itaipu Parquetec na geração automática de layouts preliminares de loteamentos habitacionais. A proposta busca auxiliar o planejamento urbano inicial por meio da criação de layouts a partir de parâmetros definidos, como dimensões de quadras, lotes e terrenos. Para isso, foi desenvolvido um conjunto de dados com 700 amostras, construído a partir de imagens geradas no AutoCAD em combinação com mapas reais da malha urbana de Foz do Iguaçu, Brasil. A metodologia envolveu o pré-processamento dos dados, a geração da malha viária (ruas), o

treinamento do modelo de Redes Adversárias Generativas (GANs) condicional e o ajuste de hiperparâmetros. A avaliação quantitativa, realizada com o Fréchet Inception Distance (FID), indicou valores de 40,61 para cenários simples e 73,09 após a inclusão da malha viária, confirmando o aumento da complexidade da tarefa. Ainda assim, a abordagem pode ser usada como ferramenta de apoio nas etapas iniciais do planejamento urbano. Como trabalhos futuros, pretende-se ampliar a base de dados para diferentes tipologias urbanas, disponibilizá-la como benchmark para a área e investigar o uso de modelos de Difusão como alternativa para melhorar a fidelidade geométrica dos layouts gerados.

Palavras-chave—Redes Adversárias Generativas; Pix2Pix; Design Generativo; Layout Residencial.

I. INTRODUÇÃO

Um loteamento consiste em uma grande área de terra subdividida em lotes menores, geralmente acompanhada de vias públicas, áreas comuns e infraestrutura básica, com o objetivo de viabilizar a construção de unidades habitacionais ou comerciais. O projeto arquitetônico de um loteamento envolve o planejamento técnico da organização do terreno, incluindo a delimitação dos lotes, o traçado das ruas e a integração de espaços públicos e equipamentos urbanos.

A elaboração de projetos dessa natureza exige conhecimento técnico especializado, atendimento à legislação urbanística vigente e compatibilização com projetos complementares, como hidrossanitário, elétrico e viário. Além disso, todo o conjunto deve ser submetido à análise e aprovação dos órgãos competentes, processo que pode se estender de seis a vinte e quatro meses, a depender da complexidade e das exigências normativas locais. Nesse contexto, ferramentas que contribuam para

otimizar etapas iniciais do planejamento urbano representam um apoio significativo para arquitetos e urbanistas, promovendo redução de custos, economia de tempo e maior padronização.

Nos últimos anos, a aplicação de técnicas de Inteligência Artificial (IA) na Arquitetura e Urbanismo tem ganhado destaque, com avanços notáveis em geração e organização de espaços. Modelos de última geração, como modelos de Difusão e *Transformers*, têm demonstrado resultados expressivos na síntese de imagens e layouts urbanos. No entanto, essas abordagens frequentemente requerem grandes volumes de dados e elevado poder computacional, o que pode dificultar sua aplicação em cenários específicos e com recursos limitados. Por outro lado, as GANs, em especial variantes condicionais como o Pix2Pix, permanecem relevantes por sua robustez, capacidade de aprendizado com conjuntos de dados menores e rapidez no treinamento, além de possibilitarem o uso de redes já pré-treinadas como ponto de partida, o que reduz ainda mais os custos de treinamento.

Diante desse cenário, este trabalho busca responder à seguinte questão de pesquisa: em que medida modelos generativos condicionais, como o Pix2Pix, conseguem gerar layouts de loteamentos residenciais coerentes do ponto de vista visual e urbanístico, a partir de dados derivados de plantas de projetos arquitetônicos de loteamentos. A hipótese central é que a adaptação de um modelo condicional para este domínio, associada à construção de um conjunto de dados específico, pode permitir a geração de propostas iniciais de loteamentos que mantenham consistência com os padrões esperados, mesmo que ainda exijam ajustes posteriores por parte de profissionais da área.

Nesse sentido, este artigo propõe uma abordagem exploratória que verifica a viabilidade e gera propostas iniciais de loteamentos a partir de imagens de entrada contendo informações urbanísticas básicas, como a configuração de quadras e lotes. A escolha do Pix2Pix justifica-se por sua adequação a cenários com quantidade limitada de dados anotados, permitindo treinar o modelo mesmo quando conjuntos de dados extensos não estão disponíveis.

As principais contribuições deste trabalho incluem: (i) a criação de um conjunto de dados de layouts de loteamentos residenciais gerado a partir de modelos CAD e mapas urbanos de Foz do Iguaçu, Brasil; (ii) a realização de uma análise empírica detalhada sobre o desempenho do Pix2Pix nesse contexto; e (iii) a identificação de limitações e potenciais direções futuras para a aplicação de IA no planejamento urbano.

O restante deste artigo está organizado como segue: a Seção II apresenta uma revisão dos modelos generativos aplicados à Arquitetura e Urbanismo, contextualizando o estado da arte e os trabalhos relacionados; a Seção III descreve a metodologia proposta, incluindo a construção do conjunto de dados, a

configuração do modelo Pix2Pix e os critérios de avaliação; a Seção IV apresenta os resultados obtidos e uma discussão sobre o desempenho do modelo, suas limitações e potencialidades; e, por fim, a Seção V traz as conclusões do trabalho e as direções para pesquisas futuras.

II. MODELOS GENERATIVOS PARA URBANISMO E ARQUITETURA

Diversos estudos aplicam variantes das GANs ao campo da arquitetura e do urbanismo. Um dos primeiros exemplos é o ArchiGAN, proposto por Chaillou [1], que utiliza três etapas sequenciais para gerar plantas de edifícios, layouts internos e disposição de mobiliário, permitindo ajustes manuais em cada fase e oferecendo maior flexibilidade ao projetista. A Figura 1 apresenta o pipeline das três etapas propostas pelo ArchiGAN. De forma semelhante, Sun et al. [2] exploram o modelo Pix2Pix para aprender preferências espaciais em layouts residenciais chineses, utilizando codificação RGB para representar diferentes elementos urbanos.

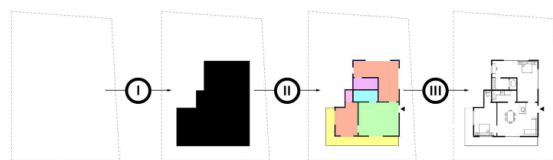


Figura 1. Pipeline do ArchiGAN com os três modelos: (I) geração de planta baixa, (II) layout interno e (III) disposição de mobiliário. Fonte: [1].

Embora esses primeiros trabalhos tenham demonstrado o potencial das GANs na geração de layouts arquitetônicos, suas limitações motivaram o desenvolvimento de abordagens mais sofisticadas. Outros trabalhos exploram condicionamentos específicos para ampliar o controle sobre os resultados. Morteza et al. [3] empregam GANs condicionais para gerar layouts a partir de representações codificadas por cores, associando cada cor a zonas com restrições geométricas e topológicas. Runjia et al. [4] utilizam Pix2PixHD para transformar limites de terrenos em representações programáticas coloridas, integrando o fluxo a ferramentas paramétricas como Rhinoceros e Grasshopper. Essas aplicações demonstram a capacidade das GANs de aprender transformações visuais complexas, mas também revelam limitações relacionadas à resolução de saída, à generalização para contextos distintos e à dependência da qualidade e diversidade dos dados de entrada.

Buscando superar essas limitações identificadas nas GANs tradicionais, estudos subsequentes introduziram novos condicionamentos e integraram dados vetoriais ao treinamento. Chen et al. [5] utilizam imagens de contorno e mapas de

centróides de cômodos como entrada para GANs, garantindo maior coerência espacial. Nelson et al. [6], [7] propõem GANs baseadas em grafos para representar relações espaciais entre cômodos, incluindo atributos como portas nas arestas, o que possibilita maior controle sobre conectividade e acessibilidade. Chen et al. [8] aplicam GANs ao design de parques urbanos, gerando alternativas baseadas em padrões paisagísticos existentes. Apesar desses avanços, o foco dessas pesquisas permanece em edificações e espaços urbanos consolidados, sem abordar fases iniciais de parcelamento do solo.

Paralelamente ao desenvolvimento das GANs, mais recentemente, modelos baseados em Difusão e *Transformers* têm redefinido o estado da arte na geração de imagens. É importante destacar que essas arquiteturas não são subtipos de GANs, mas abordagens distintas no âmbito do *Deep Learning*, com mecanismos próprios de geração. Modelos de Difusão [9] se baseiam em modelos probabilísticos explícitos, com forte fundamentação em inferência variacional e processos estocásticos, mais estáveis e teoricamente interpretáveis. Enquanto *Transformers* [10] representam uma abordagem de aprendizado de dependências globais, fundamentada em atenção e decomposição autorregressiva, generalizando a ideia de aprendizado sequencial e multimodal.

No campo do urbanismo, essas técnicas vêm sendo aplicadas para gerar paisagens urbanas, redes viárias e padrões morfológicos [11]–[14]. Resultados recentes indicam maior fidelidade visual, consistência semântica e controle refinado sobre atributos de projeto, superando limitações comuns das GANs. No entanto, esses modelos também apresentam desafios para uso prático, como elevado custo computacional, necessidade de conjunto de dados extensos e dificuldade de adaptação a dados vetoriais, como plantas CAD de loteamentos [15].

Assim, apesar dos avanços obtidos com GANs, Modelos de Difusão e *Transformers*, a literatura carece de estudos voltados à geração automática de layouts de loteamentos residenciais com conjuntos de dados urbanos reais e comparações sistêmicas entre diferentes técnicas. O presente trabalho busca preencher essa lacuna por meio da validação e análise da aplicação do modelo Pix2Pix para gerar layouts de loteamentos a partir de dados vetoriais urbanos de Foz do Iguaçu, PR. Os arquivos CAD (.dwg/.dxf) foram convertidos para imagens RGB e utilizados no treinamento da rede, visando oferecer uma ferramenta de apoio à concepção preliminar de projetos de loteamento alinhados às diretrizes urbanísticas e à morfologia local.

III. METODOLOGIA

Os experimentos foram conduzidos em um ambiente computacional baseado em *Ubuntu 22.04*, com desenvolvimento em linguagem *Python*. Para assegurar reprodutibilidade, todas

as dependências foram encapsuladas em um contêiner *Docker*, configurado especificamente para a execução dos modelos de GANs.

O hardware utilizado foi composto por um processador AMD Ryzen Threadripper 3970X (32 núcleos), uma GPU NVIDIA RTX 4500 Ada Generation. Essa configuração visou garantir desempenho adequado no processamento das imagens e no treinamento do modelo.

A metodologia foi organizada em cinco etapas principais: (i) preparação e pré-processamento do conjunto de dados, incluindo a conversão dos modelos CAD para imagens alinhadas a mapas urbanos; (ii) geração da malha viária através de morfologia matemática; (iii) treinamento; (iv) ajuste de hiperparâmetros; e (v) avaliação quantitativa dos resultados com medidas de similaridade entre imagens sintéticas e reais.

A. Criação do conjunto de imagens

Devido à ausência de bases públicas contendo imagens de loteamentos residenciais, nós criamos manualmente um conjunto de dados específico para este estudo. Para isso, utilizamos o software AutoCAD, com o suporte de um profissional da área de arquitetura, garantindo a fidelidade técnica do material.

A Figura 2 apresenta dois terrenos utilizados como referência no processo de criação do conjunto de dados. Essas áreas foram identificadas e posteriormente delimitadas pela equipe de Arquitetura e Engenharia do Itaipu Parquetec. Cada lote recebeu uma numeração (01, 02) de acordo com o projeto arquitetônico, servindo como base para a geração dos layouts de loteamentos residenciais.

O processo foi iniciado pelos autores com a importação de mapas vetoriais da cidade de Foz do Iguaçu, que serviram como referência geográfica para a delimitação dos terrenos e a distribuição das quadras. Um exemplo do mapa é apresentado na Figura 3. Para a construção das amostras, nós copiamos regiões da cidade para novos arquivos, onde as camadas e elementos foram ajustados, mantendo apenas o layout essencial dos lotes.

Cada lote foi destacado em vermelho, enquanto o fundo do terreno recebeu a cor azul. Essa escolha se deve ao alto contraste no espaço RGB, o que facilita a distinção entre lotes e terreno e contribui para que o modelo aprenda as relações espaciais de forma mais eficiente. As imagens foram elaboradas seguindo princípios de organização urbana, respeitando proporções, separação entre lotes e evitando áreas comerciais ou construções não residenciais. Essas revisões foram essenciais para reduzir inconsistências e permitir que o modelo Pix2Pix aprendesse relações espaciais coerentes.

Ao todo, foram produzidos 700 pares de imagens: a imagem de entrada, representando o terreno, e a imagem de saída, com os lotes e quadras devidamente demarcados (Figura 4).



Figura 2. Terrenos de referência utilizados como base para a criação do conjunto de dados. Os lotes (01 e 02) foram definidos e numerados pela equipe de Arquitetura e Engenharia como parte do planejamento do projeto de construção.

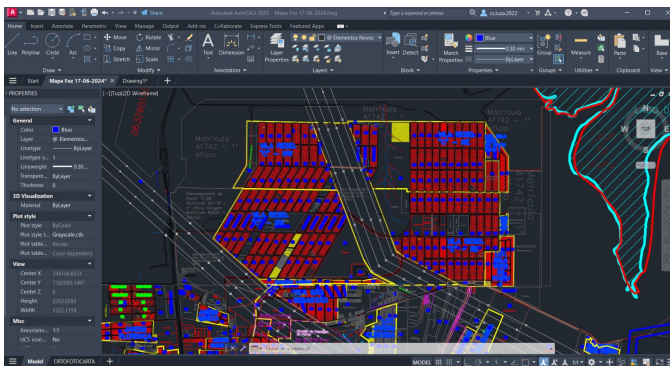


Figura 3. Trecho do mapa vetorial da cidade de Foz do Iguaçu no AutoCAD, utilizado como referência geográfica para a delimitação dos terrenos e distribuição das quadras.

B. Geração da Malha Viária

Para investigar o impacto da representação da malha viária na geração de layouts de loteamentos, testamos a hipótese de incluí-la como entrada adicional do modelo Pix2Pix. A ideia é que, ao fornecer explicitamente a estrutura viária, o modelo possa aprender relações espaciais mais complexas entre quadras e lotes, potencialmente aumentando a consistência urbanística dos layouts gerados.

Além disso, a inclusão da malha viária permite avaliar se a rede é capaz de adaptar a disposição dos lotes às restrições impostas pelas vias, aproximando os resultados de projetos reais. Essa abordagem fornece uma análise quantitativa e qualitativa

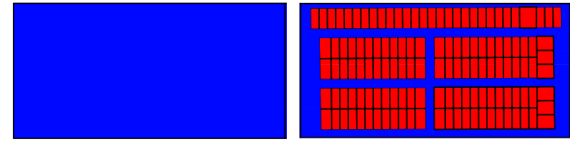


Figura 4. Exemplo de par de imagens do conjunto de dados: à esquerda, imagem de entrada com a geometria do terreno; à direita, imagem de saída com lotes e quadras demarcados.

da contribuição da informação viária para a geração automática de layouts, servindo como base para comparações com cenários em que apenas a geometria dos terrenos é fornecida como entrada.

Aplicamos técnicas de processamento de imagens, incluindo filtragem de cor, detecção de contornos e operações morfológicas de abertura e fechamento, para a geração automática da malha viária. A lógica do algoritmo segue sete etapas, conforme descrito a seguir e ilustrado na Figura 5.

- 1) **Entrada:** Nós utilizamos uma imagem do conjunto de dados original, onde o fundo azul representa o terreno e os blocos vermelhos indicam as quadras com seus respectivos lotes.
- 2) **Isolamento de cor:** Aplicamos um filtro para extrair os pixels com tonalidade vermelha, correspondentes às quadras. Em seguida, reaplicamos operações morfológicas de fechamento e abertura para correção de discontinuidades e remoção de ruídos.
- 3) **Criação de máscara:** Geramos uma máscara binária a partir dos contornos poligonais detectados nas regiões das quadras.
- 4) **Conversão de regiões:** Os pixels azuis contidos na máscara são convertidos para a cor cinza, representando as áreas destinadas às vias públicas.
- 5) **Refinamento:** Reaplicamos operações morfológicas de abertura e fechamento para consolidar as regiões convertidas, complementadas por uma filtragem por área mínima com o objetivo de remover elementos espúrios.
- 6) **Espelhamento:** As ruas geradas são replicadas simetricamente para assegurar padronização e coerência visual no conjunto de dados.
- 7) **Saída:** Produzimos uma nova imagem com as vias em cinza, delimitando de forma clara as quadras sobre o fundo azul, pronta para ser utilizada no treinamento da GAN.

C. Modelo Pix2Pix

Para a geração automática de layouts de loteamentos residenciais, adotamos o modelo Pix2Pix devido à sua eficácia comprovada em tarefas de tradução imagem-para-imagem

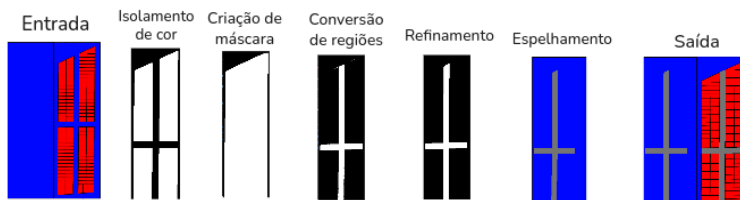


Figura 5. Fluxo das sete etapas do algoritmo para geração da malha viária, mostrando o pré-processamento, isolamento de cores, criação de máscara, conversão de regiões, refinamento, espelhamento e saída final.

condicionada [16]. Além disso, o Pix2Pix é adequado para cenários com conjuntos de dados limitados, permite utilizar redes pré-treinadas como ponto de partida, e demonstrou bom desempenho em diferentes domínios de imagens, incluindo padrões urbanos e espaciais.

Essas características tornam o modelo uma escolha apropriada para gerar layouts de loteamentos, mantendo a coerência estrutural entre quadras, lotes e ruas, mesmo em situações com informação limitada sobre a malha urbana.

As GANs são modelos capazes de aprender um mapeamento de um vetor de ruído aleatório z para uma imagem de saída y , formalizado como $G : z \rightarrow y$. Já as GANs condicionais estendem essa formulação ao aprender um mapeamento a partir de uma imagem observada x e de um vetor de ruído z , gerando uma saída y , representada por $G : x, z \rightarrow y$.

Nesse contexto, o gerador G é treinado para produzir imagens sintéticas $G(x, z)$ que sejam indistinguíveis das imagens reais y , enquanto o discriminador D é otimizado para diferenciar corretamente entre exemplos reais x, y e falsos $x, G(x, z)$ [16]. Essa interação adversarial força o gerador a aproximar sua distribuição das imagens reais, como ilustrado na Figura 6.

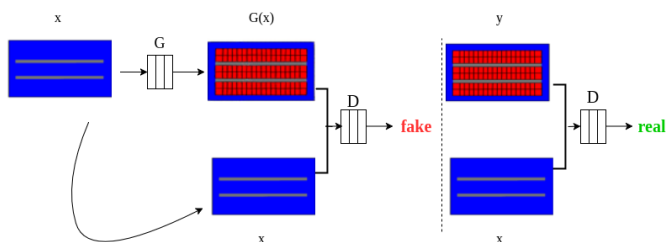


Figura 6. Arquitetura simplificada do modelo Pix2Pix, adaptada de [16]. O gerador (G) recebe como entrada a imagem condicional (por exemplo, um layout contendo ruas) e produz uma imagem sintética correspondente. O discriminador (D) avalia a consistência do par x, y , classificando-o como real (quando y pertence ao conjunto de dados) ou falso (quando $y = G(x)$ é gerado pelo modelo).

O gerador U-Net captura informações de baixa e alta frequência por meio de *skip connections*, preservando detalhes

estruturais importantes para a definição de lotes e ruas. O discriminador PatchGAN, por sua vez, avalia regiões locais da imagem, o que promove treinamento estável e maior eficiência computacional [16].

Além das vantagens arquiteturais, o Pix2Pix apresenta benefícios práticos para este estudo: (i) treinamento relativamente rápido, devido à possibilidade de utilizar redes pré-treinadas como ponto de partida e inferência ágil, adequados ao processamento de centenas de imagens derivadas de modelos CAD; (ii) preservação da correspondência espacial entre os elementos de entrada (terrenos e lotes) e saída (layouts completos); e (iii) ampla disponibilidade em bibliotecas de *Deep Learning*, com documentação consolidada que favorece a reprodutibilidade.

Modelos mais recentes, como Redes de Difusão, ainda que capazes de gerar imagens de maior diversidade e qualidade, exigem conjuntos de dados e infraestrutura significativamente mais robustos, o que os torna menos viáveis nesta etapa de pesquisa.

A subseção a seguir detalha o ajuste de hiperparâmetros, a arquitetura do gerador e do discriminador, bem como demais configurações utilizadas no treinamento do modelo.

D. Ajuste de Hiperparâmetros

O ajuste de hiperparâmetros é uma etapa crítica no treinamento de GANs. A seleção adequada desses parâmetros influencia diretamente a estabilidade do treinamento, a diversidade das amostras geradas, a fidelidade visual dos resultados e o custo computacional. A exploração exaustiva do espaço de busca é inviável na prática, o que exige estratégias sistemáticas para identificar combinações eficazes de forma mais rápida.

Problemas recorrentes, como o *mode collapse* e a não convergência, causados pela dinâmica competitiva entre gerador e discriminador, são frequentemente atribuídos à má configuração dos hiperparâmetros [17]. Por esse motivo, a escolha dos parâmetros foi orientada por três critérios: (i) diretrizes presentes na literatura técnica, especialmente em [16]; (ii) boas práticas de repositórios e benchmarks públicos; e (iii) observações empíricas de treinamentos preliminares.

Dentre os hiperparâmetros avaliados, destacam-se:

- **beta1**: parâmetro de momento do otimizador Adam, testado com valores 0,5 e 0,9. Ambos mostraram boa convergência.
- **gan_mode**: define a função de perda adversarial. Testaram-se os modos *vanilla*, *lsgan* e *wgangp*. O *lsgan* apresentou melhor desempenho geral e maior convergência.
- **init_type**: tipo de inicialização dos pesos: *normal*, *xavier*, *kaiming* e *orthogonal*. Os melhores resultados visuais e quantitativos ocorreram com *normal* e *kaiming*.
- **lambda_L1**: peso da penalização L1 na função de custo. Dentre os valores testados, 25 e 100 proporcionaram equilíbrio na fidelidade visual, mantendo a similaridade com as imagens de referência.
- **netG e netD**: arquiteturas da rede geradora e discriminadora, respectivamente. A combinação *unet_128* e *basic* apresentou melhor desempenho geral, produzindo layouts com maior coerência estrutural, menor ruído e convergência estável.
- **ngf, ndf**: número de filtros nas primeiras camadas. O valor de 128 filtros foi adotado por oferecer um equilíbrio adequado entre expressividade do modelo e eficiência computacional.

A Tabela I resume os valores testados e os critérios adotados na escolha final. Os experimentos foram realizados variando um ou dois hiperparâmetros por vez, com os demais mantidos constantes. Métodos automatizados de busca de hiperparâmetros não foram aplicados devido ao tempo elevado necessário para retreinar e testar o modelo a cada alteração de configuração. Cada execução completa do Pix2Pix requer horas de treinamento, o que tornou inviável explorar exaustivamente todas as combinações possíveis de parâmetros.

A Figura 7 ilustra esse procedimento, representando o ciclo iterativo de treinamento, avaliação de desempenho e ajuste dos hiperparâmetros até a seleção do conjunto mais adequado.

Nos primeiros experimentos, buscou-se otimizar a arquitetura do gerador (*netG*), discriminador (*netD*) e o número de filtros iniciais. A configuração *unet_128* com *basic* e 128 filtros resultou em uma redução de aproximadamente 30 unidades no FID e foi adotada como base.

Nas etapas seguintes, foram avaliados os hiperparâmetros *beta1*, *lambda_L1*, *gan_mode* e *init_type*. O uso de *gan_mode = lsgan* resultou em melhoria de 12 unidades no FID. O modo *wgangp*, por sua vez, apresentou desempenho inferior. Em linha com as recomendações de [18], foram testados valores de *lambda_L1* igual a 10, 25, 50 e 100, sendo 25 e 100 os mais eficazes.

Não foi identificada uma combinação ótima de hiperparâmetros para *beta1*, pois seu desempenho dependia da inte-

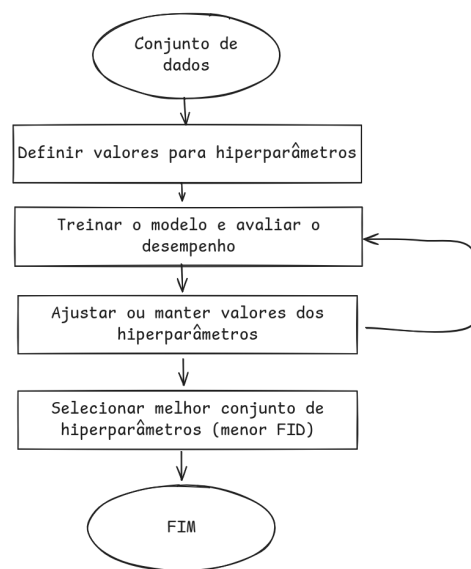


Figura 7. Fluxo do processo de ajuste de hiperparâmetros, destacando as etapas de definição inicial, treinamento, avaliação iterativa e seleção do conjunto final. As setas representam a progressão das decisões ao longo do processo.

ração com os demais parâmetros. Essa dependência reforça a importância de testes controlados com avaliação gradual de combinações específicas.

Para o conjunto de dados com a malha viária, nós optamos por reduzir o número de experimentos, priorizando os conjuntos de hiperparâmetros que previamente apresentaram resultados promissores. Nessa etapa, o foco foi a variação dos hiperparâmetros *beta1*, *init_type* e *lambda_L1*. Adicionalmente, foram conduzidos dois experimentos utilizando a função de perda *vanilla*, com o intuito de comparar seu desempenho com o *gan_mode (lsgan)* adotado anteriormente.

E. Fréchet Inception Distance

A medida FID, introduzida por [19], é amplamente utilizada para avaliar a qualidade de imagens sintéticas por modelos baseados em GANs. Essa medida compara as distribuições estatísticas dos vetores de características extraídos de imagens reais e sintéticas, utilizando para isso o modelo Inception-v3 proposto por [20].

Neste trabalho, o FID foi calculado a partir de um conjunto de teste contendo 145 imagens. Os vetores de características foram extraídos de uma camada intermediária do Inception-v3, e considerados como distribuições gaussianas multivariadas. A dissimilaridade entre essas distribuições é então quantificada por meio da distância de Fréchet, que incorpora tanto a diferença entre as médias quanto a divergência entre as matrizes de covariância.

Tabela I
RESUMO DOS HIPERPARÂMETROS TESTADOS E CRITÉRIOS UTILIZADOS PARA A ESCOLHA FINAL.

Hiperparâmetro	Valores testados	Critério de escolha final
beta1	0,5; 0,9	Ambos apresentaram boa estabilidade durante o treinamento
gan_mode	vanilla; lsgan; wgangp	lsgan proporcionou curvas de perda mais suaves e melhor generalização
init_type	normal; xavier; kaiming; orthogonal	normal e kaiming facilitaram a convergência
lambda_L1	10; 25; 50; 100	Valores 25 e 100 resultaram em melhor equilíbrio entre fidelidade visual e suavidade
netG	unet_128; unet_256; resnet_6blocks	unet_128 apresentou melhor qualidade visual e treinamento mais rápido
netD	basic; n_layers; pixel	basic demonstrou discriminação eficaz
ngf, ndf	32; 64; 128	128 proporcionou maior capacidade de aprendizado com custo computacional viável

Valores mais baixos de FID indicam maior similaridade estatística entre as representações de imagens reais e sintéticas. Em um cenário ideal, o FID tende a zero, o que implicaria que as duas distribuições são idênticas. Na prática, pontuações FID reduzidas estão associadas a imagens sintéticas com maior realismo e fidelidade morfológica, considerando aspectos como forma, proporção e densidade das estruturas visuais [20].

A equação que define o FID [19] é dada por:

$$FID = ||\mu_r - \mu_g||^2 + Tr(\Sigma_r + \Sigma_g - 2\sqrt{\Sigma_r \cdot \Sigma_g}) \quad (1)$$

onde μ_r e μ_g representam os vetores de média das características extraídas das imagens reais e sintéticas, respectivamente; e Σ_r e Σ_g são as correspondentes matrizes de covariância. Esses parâmetros são geralmente calculados a partir de vetores de 2048 dimensões, fornecendo uma representação estatística compacta das imagens analisadas [21].

IV. RESULTADOS E DISCUSSÃO

Durante os experimentos iniciais, nós observamos a ocorrência de *mode collapse*, fenômeno em que o gerador passa a produzir amostras com baixa diversidade, comprometendo a generalização do modelo. Esse comportamento foi evidenciado pela persistência de altos valores na perda G_{L1} e pela aproximação dos valores de D_{real} e D_{fake} a zero, como ilustrado na Figura 8. Esse padrão indica um desbalanceamento entre gerador e discriminador, sugerindo que o discriminador estava excessivamente dominante na dinâmica adversarial.

Com base nessas observações, diferentes configurações de hiperparâmetros foram avaliadas. A Figura 9 mostra a evolução da perda G_{L1} com o Conjunto 1 da Tabela II, evidenciando uma convergência mais estável e uma dinâmica adversarial mais equilibrada em comparação ao treinamento inicial.

A Tabela III resume os valores finais das medidas adversariais para dois treinamentos: o inicial, com hiperparâmetros padrão, e o otimizado, com o Conjunto 1. Houve redução expressiva na perda G_{L1} (de 29,005 para 6,35) e aumentos nos

Tabela II
CONJUNTOS DE HIPERPARÂMETROS COM MELHOR DESEMPENHO.

Hiperparâmetros	Conjunto 1	Conjunto 2	Conjunto 3	Conjunto 4
beta1	0,9	0,5	0,5	0,9
gan_mode	lsgan	lsgan	lsgan	lsgan
init_type	kaiming	normal	kaiming	kaiming
lambda_L1	100	25	25	25
netG	unet_128	unet_128	unet_128	unet_128
netD	basic	basic	basic	basic
ngf	128	128	128	128
ndf	128	128	128	128

valores de D_{real} e D_{fake} , sugerindo melhor equilíbrio entre os componentes da rede adversarial.

Tabela III
VALORES FINAIS DAS MEDIDAS ADVERSARIAIS ENTRE O TREINAMENTO INICIAL E O OTIMIZADO.

Medida	Inicial	Otimizado
G_{L1}	29,005	6,35
D_{real}	0,006	0,128
D_{fake}	0,003	0,160

A Tabela II apresenta os quatro conjuntos de hiperparâmetros que resultaram nos menores valores de FID. Os Conjuntos 1 e 2 foram mais eficazes no treinamento com o conjunto de dados inicial (sem ruas), enquanto os Conjuntos 3 e 4 apresentaram melhor desempenho com a inclusão da malha viária.

A inicialização dos pesos mostrou-se relevante para a dinâmica de aprendizado. Modelos com *init_type* = *kaiming* demandaram maior número de épocas para convergência, apresentando melhor desempenho após treinamento prolongado. Já com *init_type* = *normal*, a convergência foi mais rápida, sendo 300 épocas suficientes para resultados satisfatórios.

Com o conjunto de dados inicial, o menor FID registrado foi 40,61. A análise visual indicou que o modelo foi capaz

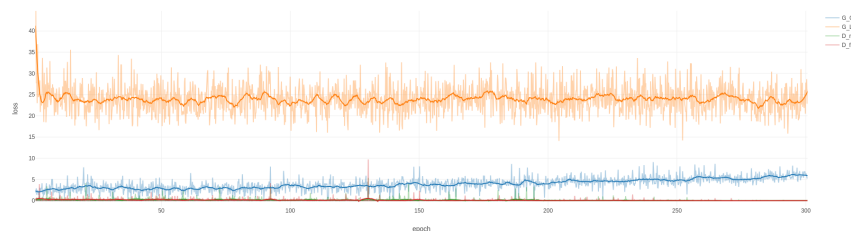


Figura 8. Curvas de perda durante o treinamento inicial com hiperparâmetros padrão. A linha laranja corresponde à perda G_{L1} , a azul à perda G_{GAN} , enquanto as linhas vermelha e verde (próximas de zero) representam D_{real} e D_{fake} , respectivamente. Observa-se ocorrência de *mode collapse*, evidenciada pela oscilação irregular das curvas e ausência de convergência estável.

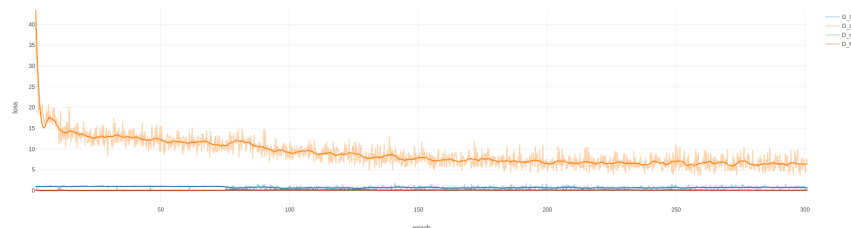


Figura 9. Curvas de perda durante o treinamento com o Conjunto 1 da Tabela II. As cores seguem a mesma convenção da Figura 8. Neste caso, nota-se convergência estável das funções de perda, com redução mais acentuada de G_{L1} e comportamento consistente de G_{GAN} , indicando treinamento adequado do modelo.

de representar de forma satisfatória os formatos e distribuições dos lotes, especialmente em cenários de menor complexidade. No entanto, o valor ainda elevado de FID reflete inconsistências nas imagens sintéticas. Já com o conjunto contendo ruas, o FID foi de 73,09, indicando maior dificuldade da GAN em aprender a estrutura urbana.

As Figuras 10 e 11 mostram amostras de saída da GAN em ambos os conjuntos. As imagens sintéticas apresentam características estruturais coerentes com as referências, mas ainda exibem limitações em casos mais complexos.

Apesar dos avanços, o modelo apresentou falhas na geração de layouts atípicos ou mais complexos, como ilustrado nas Figuras 12 e 13. Nesses casos, as saídas diferem significativamente da imagem alvo.

Como possibilidade de aprimoramento, recomenda-se ampliar a base de dados com exemplos mais variados de configurações urbanas e aplicar técnicas de ajuste fino específicas para elementos urbanos, como vias e interseções.

V. CONCLUSÃO

Este trabalho apresentou um estudo exploratório sobre a aplicação do modelo Pix2Pix para a geração automática de layouts de loteamentos residenciais. Para isso, foi desenvolvido um conjunto de dados específico, combinando imagens CAD e o mapa de Foz do Iguaçu, e realizada uma análise comparativa entre dois cenários: com e sem a inclusão das ruas como entrada

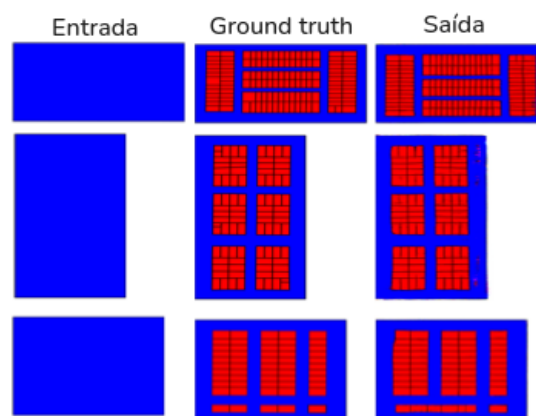


Figura 10. Exemplos de imagens sintéticas (Saída) pela GAN.

do modelo. O trabalho buscou atender a uma necessidade prática da equipe de Arquitetura e Engenharia envolvida na realocação de famílias em condições de vulnerabilidade, oferecendo suporte na elaboração inicial de projetos de parcelamento do solo.

Os resultados indicaram que, no cenário sem ruas, o FID foi de 40,61, mostrando que o modelo foi capaz de gerar layouts coerentes com a distribuição de lotes de forma satisfatória,

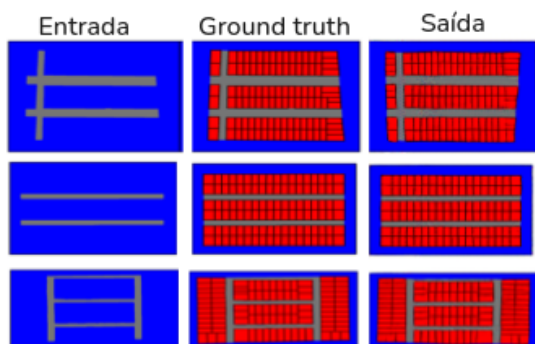


Figura 11. Exemplos de imagens sintéticas (Saída) pela GAN com a malha viária.

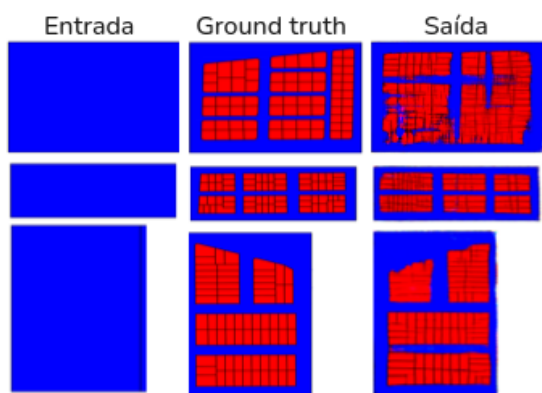


Figura 12. Exemplos de imagens sintéticas com falhas no conjunto de dados inicial.

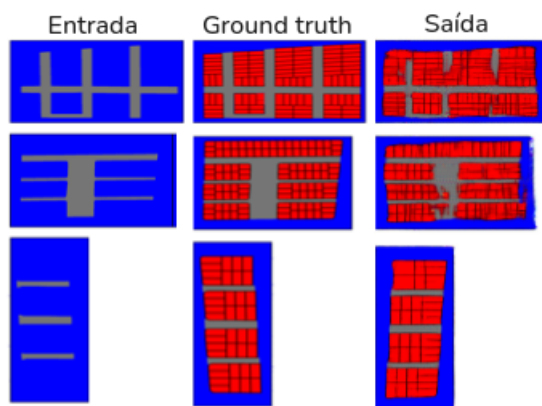


Figura 13. Exemplos de imagens sintéticas com falhas no conjunto de dados com ruas.

especialmente em cenários de menor complexidade. Quando as ruas foram incluídas, o FID aumentou para 73,09, sugerindo maior dificuldade do modelo em capturar a estrutura urbana mais complexa. Ainda assim, a presença das ruas contribuiu para uma melhor consistência espacial e organização dos lotes, evidenciando o valor dessa abordagem nas etapas iniciais do planejamento urbano.

A diferença de desempenho entre os cenários pode estar relacionada à complexidade adicional do layout, à exploração limitada de hiperparâmetros ou ao tamanho relativamente reduzido do conjunto de dados (700 amostras), que pode ter limitado a capacidade de generalização do modelo. Embora esses fatores não tenham sido totalmente isolados neste estudo, os resultados fornecem indícios importantes sobre o impacto da inclusão das ruas na geração de layouts.

Como desdobramentos futuros, sugerimos: a aplicação de modelos de última geração, como redes de Difusão, a ampliação do conjunto de dados com exemplos mais variados e elementos urbanos adicionais e/ou a inclusão de parâmetros adicionais para guiar a geração, permitindo resultados mais adaptáveis a diferentes restrições urbanísticas.

Além disso, embora os resultados visuais obtidos já demonstrem potencial para apoiar a etapa inicial de concepção urbanística, a utilização prática ainda exigiria um pós-processamento vetorial para corrigir inconsistências geométricas, como sobreposições, descontinuidades ou erros topológicos nos limites dos lotes.

Outra linha promissora seria o desenvolvimento de novas arquiteturas de redes generativas capazes de modelar diretamente as estruturas espaciais dos lotes, e não apenas sua representação em pixels. Abordagens dessa natureza poderiam capturar relações topológicas e métricas de forma mais precisa, aproximando o modelo de aplicações reais em planejamento urbano assistido por IA.

AGRADECIMENTOS

Os autores agradecem à equipe de Arquitetura & Engenharia do Itaipu Parquetec pelo apoio essencial no desenvolvimento deste projeto, contribuindo com conhecimento técnico e colaboração ao longo de todo o processo. Expressamos nossa gratidão ao Itaipu Parquetec, pelo suporte institucional e pela oportunidade de realizar este trabalho em um ambiente de constante inovação e aprendizado.

REFERÊNCIAS

- [1] S. Chaillou, "Archigan: Artificial intelligence x architecture," in *Architectural Intelligence: Selected Papers from the 1st International Conference on Computational Design and Robotic Fabrication (CDRF 2019)*. Springer, 2020, pp. 117–127. [Online]. Available: https://doi.org/10.1007/978-981-15-6568-7_8

- [2] P. Sun, F. Yan, Q. He, and H. Liu, "The development of an experimental framework to explore the generative design preference of a machine learning-assisted residential site plan layout," *Land*, vol. 12, no. 9, p. 1776, 2023.
- [3] M. Rahbar, M. Mahdavejad, M. Bemanian, A. H. D. Markazi, and L. Hovestadt, "Generating synthetic space allocation probability layouts based on trained conditional-gans," *Applied Artificial Intelligence*, vol. 33, no. 8, pp. 689–705, 2019.
- [4] R. Tian, "Suggestive site planning with conditional gan and urban gis data," in *Proc. 2020 DigitalFUTURES (CDRF)*, P. F. Y. et al., Ed., 2021, pp. 103–113.
- [5] A. Chen, "Generation of layouts for living spaces using conditional generative adversarial networks: Designing floor plans that respect both a boundary and high-level requirements," arXiv preprint, 2022. [Online]. Available: <https://arxiv.org/abs/2207.05669>
- [6] N. N. et al., "House-gan++: Generative adversarial layout refinement network towards intelligent computational agent for professional architects," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 13 632–13 641.
- [7] —, "House-gan: Relational generative adversarial networks for graph-constrained house layout generation," in *Proc. Eur. Conf. on Computer Vision (ECCV)*, vol. 12346. Springer, 2020, pp. 162–177.
- [8] R. C. et al., "Generative design of outdoor green spaces based on generative adversarial networks," *Buildings*, vol. 13, no. 4, p. 1083, 2023.
- [9] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.
- [10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- [11] K. Xu, Y. Li, and H. Wang, "Urbangenai: Reconstructing urban landscapes using panoptic segmentation and diffusion models," arXiv preprint, 2024. [Online]. Available: <https://arxiv.org/abs/2401.14379>
- [12] J. Smith, L. Chen, and M. Zhao, "Generating urban road networks with conditional diffusion models," *ISPRS International Journal of Geo-Information*, vol. 13, no. 6, p. 203, 2024. [Online]. Available: <https://www.mdpi.com/2220-9964/13/6/203>
- [13] A. Li, P. Zhang, and R. Wang, "Conditional diffusion model for urban morphology prediction," *Remote Sensing*, vol. 16, no. 10, p. 1799, 2024. [Online]. Available: <https://www.mdpi.com/2072-4292/16/10/1799>
- [14] Z. Qian, M. Li, and R. Zhang, "Dalle-urban: Capturing the urban design expertise of large text-to-image transformers," arXiv preprint, 2022. [Online]. Available: <https://arxiv.org/abs/2208.04139>
- [15] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," arXiv preprint, 2020. [Online]. Available: <https://arxiv.org/abs/2006.11239>
- [16] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5967–5976.
- [17] D. Hölscher, C. Reich, F. Gut, M. Knahl, and N. L. Clarke, "Pix2pix hyperparameter optimisation prediction," *Procedia Computer Science*, vol. 225, pp. 1009–1018, 2023.
- [18] J.-Y. Zhu, "Can you please describe about the gan mode in details?" GitHub issue #1011, pytorch-CycleGAN-and-pix2pix, 2020, accessed: 2025-05-20. [Online]. Available: <https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix/issues/1011>
- [19] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 6626–6637.
- [20] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 2818–2826.
- [21] T. Oanda, "A review of the image quality metrics used in image generative models," Paperspace Blog, 2023,

accessed: 2025-05-07. [Online]. Available: <https://blog.paperspace.com/review-metrics-image-synthesis-models/>