

Jonlenes: Colocar um titulo em pt



Universidade Estadual de Campinas
Instituto de Computação



Jonlenes Silva de Castro

**Improving Seismic Data with Generative Adversarial Networks
for Horizon Tracking**

Orientadora: Profa. Dra. Sandra Eliza Fontes de Avila
Coorientador: Prof. Dr. Edson Borin

CAMPINAS
2018

Resumo

A partir de distúrbios físicos artificiais são obtidos os dados sísmicos, responsáveis pelo fornecimento de informações sobre as características do ambiente geológico. Estas que após serem processadas são utilizadas na interpretação, sendo identificadas as características do subsolo e sua localização. Ainda durante uma das etapas da interpretação, realiza-se o rastreamento de horizontes que visa identificar os limites de cada subsuperfície. A complexidade desta tarefa é definida pelos artefatos presentes no dado obtido, tal como por distorções presentes nos dados, provocadas pela proximidade com falhas ou zonas geologicamente complicadas. Neste trabalho propomos a melhoria dos dados sísmicos através da remoção de artefatos, que melhorará a precisão das estruturas geológicas, podendo beneficiar diversas aplicações, tais como, o rastreamento de horizontes. Nossa proposta consiste no desenvolvimento de modelos de Deep learning capazes de estimar um dado sísmico nítido (sem artefatos) a partir de dados com artefatos. Especificamente, propomos a construção de uma Generative Adversarial Networks que será utilizada no treinamento de dados sintéticos e aplicada aos dados reais. Os dados sintéticos com artefatos serão modelados através da convolução de uma estrutura como a Point Spread Function extraída de dados sísmicos reais de difração e dos dados nítidos, por meio da convolução desta estrutura com a Ricker wavelet. Por fim, tem se como objetivo para a rede ser capaz de aprender a PSF reversa, conseguindo assim, remover os artefatos.

Sumário

1	Introdução	4
1.1	Objetivos	5
1.2	Organização do Trabalho	5
2	Conceitos Relacionados	6
2.1	Generative Adversarial Networks	6
2.1.1	Adversarial Game	6
2.1.2	Arquitetura da Deep Convolutional GAN	7
2.1.3	Problemas e Desafios	8
2.1.4	Propostas de GANs	9
2.2	Experimento e aquisição sísmica	10
2.3	Rastreamento de horizontes	12
3	Trabalhos relacionados	15
3.1	Horizon Tracking	15
3.2	GANs	15
4	Metodologia e Método	17
4.1	Dataset	18
4.2	Cronograma	18
5	Experimentos e Resultados	20
5.1	GANs	20
5.2	Sísmica	21
6	Conclusão	24

Capítulo 1

Introdução

A caracterização de uma subsuperfície é realizada ao ser obtido dados das estruturas geológicas de um ambiente. Para isto, realiza-se a aquisição de dados sísmicos, que ao serem processados, poderão ser analisados. A interpretação destes dados é responsável por possibilitar a detecção de estruturas geológicas, tais como o petróleo. Dentre as etapas da interpretação está o rastreamento de horizontes, uma técnica utilizada para identificar a estrutura geológica da subsuperfície, que possibilita tanto a sua reconstrução quanto a de modelos estratigráficos [27]. No entanto, como mencionado por alguns autores [14, 26, 27, 30], a imprecisão em dados sísmicos — que pode ocorrer em zonas geologicamente complicadas, próximas de falhas ou por erros no processo de aquisição — é um dos principais fatores que impossibilita a correta realização do rastreamento.

Com o intuito de obter maior eficiência, diferentes métodos para o rastreamento foram desenvolvidos destacando-se o *auto tracking*. Por meio desta abordagem o intérprete, com o auxílio de um algoritmo, seleciona pontos iniciais de entrada para que seja realizada a busca de características semelhantes na vizinhança. Este algoritmo, mantém-se em execução de acordo com a resposta obtida pelos critérios específicos dos traços sísmicos. Devido a isso, a presença de artefatos podem provocar falhas no processo de rastreamento, impactando no ciclo de tempo deste processo [7, 27].

Objetivando minimizar estas imprecisões e prover melhorias em diversas aplicações, tais como o rastreamento de horizontes, propomos um modelo de *Deep Learning* para realizar o processo de remoção de artefatos em dados sísmicos migrados em tempo, na tentativa de aproximar-se de uma aquisição perfeita (modelada com a Ricker wavelet [11]). Nesse caso, será necessário desenvolver uma *Deep Neural Network* (DNN) capaz de aprender a remover apenas artefatos e manter a integridade dos dados. Para isso, propomos o desenvolvimento de uma **Generative Adversarial Networks** (GAN), DNN estado da arte para criação de amostras sintéticas e tradução de imagem para imagem, sendo esta última a utilizada neste trabalho através do mapeamento de imagens com artefatos para imagens nítidas.

Especificamente, nossa proposta é modelar dados sísmicos sintéticos utilizando o Deblurring, técnica de remoção de artefatos em imagens com desfoque. Este desfoque pode ser modelado como uma convolução de uma Point Spread Function (PSF) com uma imagem nítida. Nosso método consiste na criação de estruturas, em que será aplicada a Ricker wavelet para a geração de dados nítidos sintéticos e aplicada uma PSF extraída de

dados reais de difração, que representam os dados com artefatos. Essa PSF foi escolhida com a finalidade de obtermos uma melhor representação dos dados reais nos nossos dados sintéticos. Em assim sendo, desenvolveremos GANs que serão treinadas com esses dados e, posteriormente, aplicada nos dados reais.

A abordagem proposta por este trabalho, terá como base o que foi realizado por Lu et al. [16], em que uma GAN foi treinada e conseguiu aprender um filtro capaz de aumentar a resolução de dados sísmicos sem perder suas propriedades principais (amplitude e frequência). De modo similar, será realizado o treinamento de uma GAN com enfoque em aprender como remover artefatos, sem mudanças significativas nas propriedades sísmicas dos dados. Os objetivos propostos por este trabalho, estão apresentados a seguir.

1.1 Objetivos

O principal objetivo deste trabalho consiste em desenvolver modelos de *Deep Learning* para o aprimoramento de dados sísmicos em geral, através da remoção de artefatos. Podemos resumir nossos objetivos como:

- Avaliação de diferentes arquiteturas de GANs: várias GANs serão avaliadas a fim de selecionar a que mais se adéqua a proposta deste trabalho;
- Proposta de novas arquiteturas: dependendo dos resultados obtidos na avaliação das arquiteturas, uma nova arquitetura será proposta;
- Remoção de artefatos em dados sísmicos sintéticos: realizar o treinamento da rede para remover artefatos, inicialmente em dados sintéticos (Conforme Capítulo 4);
- Aplicação em dados reais: após a criação de um modelo robusto, será realizada sua aplicação em dados reais visando remover artefatos;
- Validação dos resultados no rastreamento de horizontes: análise comparativa entre o rastreamento de horizontes utilizando dados originais e com dados filtrados, auxiliada pela ferramenta *OpenDetect* (sistema de software de interpretação sísmica para processamento, visualização e interpretação de dados sísmicos de vários volumes¹).

1.2 Organização do Trabalho

Este trabalho está organizado da seguinte maneira. No Capítulo 2, apresentamos alguns conceitos: GANs — experimentos, aquisição e dados sísmicos – e rastreamento de horizontes. No Capítulo 3, revisamos a literatura de rastreamento de horizonte e melhoramento de dados sísmicos com GANs. Já no Capítulo 4, descrevemos as etapas gerais para alcançar nossos objetivos, dentre elas o *dataset* e a linha do tempo. No Capítulo 5 apresentamos alguns experimentos realizados e seus resultados preliminares. Finalmente no Capítulo 5, apresentamos as conclusões.

¹<https://www.dgbes.com/>

Capítulo 2

Conceitos Relacionados

Neste capítulo são apresentados alguns conceitos relacionados a este trabalho. A Seção 2.1 introduz o conceito de GANs, suas arquiteturas, problemas e desafios. Na Seção 2.2, há os processos de experimentação e aquisição sísmica realizados para a coleta de dados e, finalmente, na Seção 2.3 apresentamos o rastreamento de horizontes.

2.1 Generative Adversarial Networks

O conceito de GANs foi proposto por Goodfellow et al. [10] em 2014 como sendo um método capaz de produzir amostras sintéticas semelhantes as de um conjunto de dados, tendo sido usado para gerar novos conteúdos, como música, sons e imagens, dada à capacidade das GANs de aprenderem uma distribuição e criar uma nova amostra desta [3].

Nas GANs são treinadas duas DNN: o gerador (G) que tenta transformar um ruído de entrada em uma amostra realista e um discriminador (D) que tenta identificar amostras falsas, competindo entre si em um jogo contraditório [3] (Subseção 2.1.1). Mais especificamente, o discriminador recebe os dados reais do conjunto de treinamento e os dados falsos do gerador, classificando cada uma das entradas em real ou falso. Por outro lado, o gerador recebe o ruído e tenta gerar amostras que possam enganar o discriminador. Este processo está representado na Figura 2.1.

O modelo generativo pode ser considerado análogo a um falsificador de moeda que tenta não ser detectado, enquanto o modelo discriminativo é análogo à polícia que tenta detectar a moeda falsa. Esta competição leva ambos (G e D) a melhorar seus métodos até que as falsificações sejam indistinguíveis [10]. Quando treinados corretamente, tanto o gerador quanto o discriminador chegam a um equilíbrio de Nash [17], onde não é possível um encontrar vantagens sobre o outro [3].

2.1.1 Adversarial Game

As GANs baseiam-se no jogo não cooperativo de soma zero (ou minmax), em que enquanto um oponente quer maximizar suas próprias ações o outro quer minimizá-las. Neste caso, as redes (G e D) competem em um jogo de dois jogadores com a função $V(G, D)$:

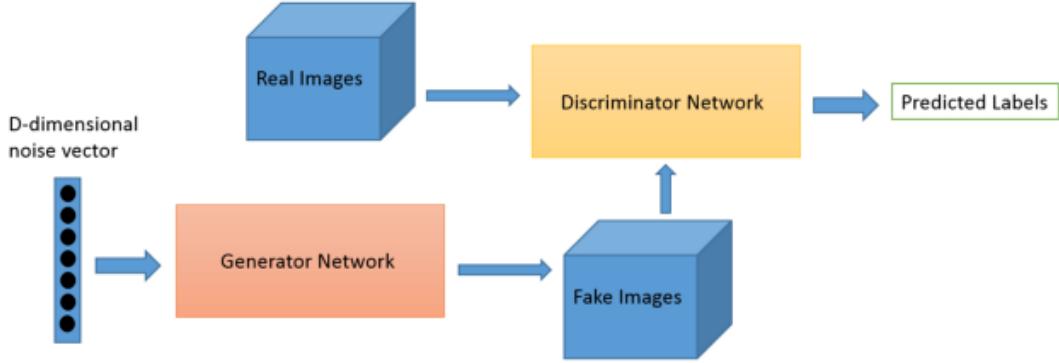


Figura 2.1: Representação das GANs: o gerador (*Generator Network*) recebe como entrada um ruído (*noise vector*) que é convertido em imagens falsas (*Fake Images*). O discriminador (*Discriminator Network*) recebe as imagens reais (*Real Images*) e as imagens falsas e realizada a classificação de cada uma delas em real ou falsa (*Predicted Labels*). Figura reproduzida de Carey [5].

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))] , \quad (2.1)$$

onde z é o ruído, p_z é a distribuição de ruído, x é o dado real e p_{data} é a distribuição de dados reais. Um equilíbrio de Nash acontece quando um jogador não muda sua ação independentemente da ação do oponente.

2.1.2 Arquitetura da Deep Convolutional GAN

Nos últimos anos diversas arquiteturas de GANs foram propostas, entretanto, a maioria delas baseiam-se em *Deep Convolutional GAN* (DCGAN) para imagens proposta por Radford et al. [20] em 2015. A arquitetura geral da DCGAN é apresentada na Figura 2.2, onde G e D representam DNNs, $P_z(z)$ é a distribuição de ruído e X é o conjunto de dados com amostras reais. O gerador recebe um vetor de ruído aleatório (z) e cria uma saída $G(z)$ que é passada para o discriminador. O discriminador determina a probabilidade da imagem ser real (p).

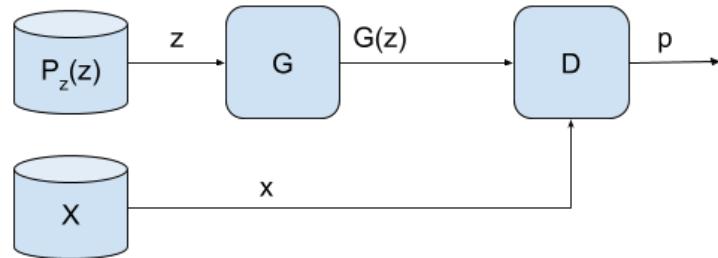


Figura 2.2: Arquitetura geral do treinamento adversarial.

Gerador e Discriminador

A entrada do gerador é um vetor de ruído aleatório. Este é repassado para a primeira camada da rede, chamada de *project and reshape*, que expande os dados de entrada. Em seguida temos os blocos convolucionais que duplicam o tamanho do primeiro e segundo eixo, e diminui gradualmente o número de canais até a quantidade desejada de canais na imagem de saída. Essas camadas convolucionais são usadas realizar *up-sampling*, que alguns autores chamam de deconvolução. Já arquitetura do discriminador é semelhante às redes de classificação e pode ser vista como uma rede que faz as operações opostas do gerador, sendo o resultado resumido a uma classificação: real ou falsa. Na Figura 2.3 é apresentado um exemplo de arquitetura DCGAN.

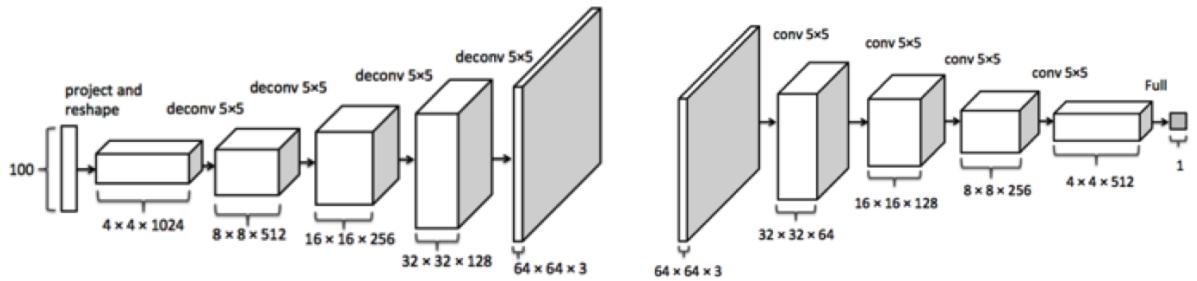


Figura 2.3: Exemplo de arquitetura do gerador (esquerda) e do discriminado. Figura reproduzida de Zhang [31].

2.1.3 Problemas e Desafios

Segundo Arjovsky and Bottou [1], as GANs são instáveis e difíceis de treinar, sendo ainda relatado por Bernico [3] que no treinamento de uma GAN existem inúmeras possibilidades de falhas. Melhorar essas características é importante para evitar problemas com o modelo e seus resultados.

Dentre estes problemas está o balanceamento entre o gerador e o discriminador, em que apesar de estarem lutando entre si, eles ainda precisam um do outro para melhorar. Sendo, para tanto, recomendado por Bernico [3] o ajuste das arquiteturas e hiperparâmetros a fim de evitar que um domine o outro.

Já o colapso do modo, um problema de difícil resolução em GANs [1, 2], caracteriza-se como sendo a produção de amostras de baixa variedade pelo gerador [24]. Nesse caso, o gerador explora um único aspecto da distribuição de treinamento, de modo a enganar o discriminador produzindo imagens similares as originais. Um exemplo deste problema pode ser visto na Figura 2.4.

Além desses, a seleção de hiperparâmetros também é um desafio, pois, em rede neurais a escolha destes são frequentemente auxiliadas pelos valores da função de custo que se deseja minimizar, entretanto, as GANs utilizam-se de uma perda relativa entre adversários, implicando que a minimização das funções de custo de G e D pode não ser a solução ótima para o problema, dificultando a escolha dos hiperparâmetros mais adequados.

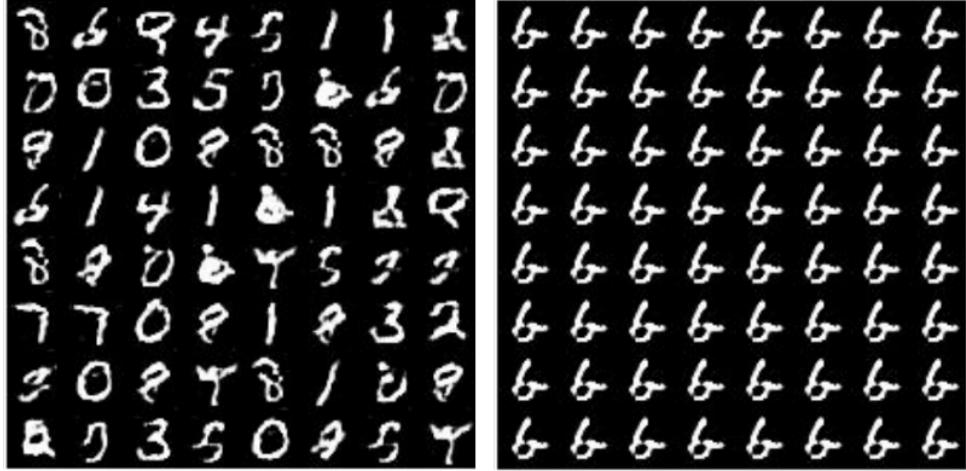


Figura 2.4: A esquerda, o modelo produz todos os 10 modos do *dataset* MNIST, enquanto a direita é criada apenas um modo. Figura adaptada de Metz et al. [18].

2.1.4 Propostas de GANs

Diversos trabalhos propuseram melhorias visando a diminuição de problemas presentes nas GANs além da ampliação da sua utilização. Ainda em 2014, Osindero [19] introduziu a *conditional Generative Neural Network* (cGANs), que estendem as GANs a um modelo condicional, onde uma entrada adicional c é adicionada em ambas as redes (G e D) [28], conforme mostrado na Figura 2.5. Essa entrada adicional c poderia ser qualquer tipo de informação auxiliar, como rótulos de classe. Logo, a nova função objetivo do jogo minimax será:

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x|c)} [\log D(x)] + E_{z \sim p_z(z)} [\log(1 - D(G(z|c)))] , \quad (2.2)$$

onde c representa a entrada adicional.

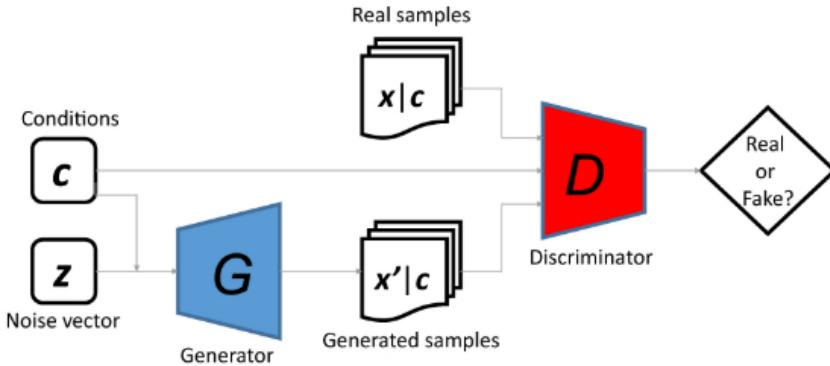


Figura 2.5: Arquitetura da cGAN: a arquitetura geral segue a mesma proposta para as GANs inicialmente, entretanto, G também deverá aprender o dado adicional c , assim como G o utilizará para classificar um exemplo em real ou falso [28].

Em 2016, Salimans et al. [23] propôs o uso de *minibatch discrimination* e de *feature mapping* para lidar com o colapso de modo e instabilidade. No *minibatch discrimination*,

o discriminador considera todo o lote para decidir se é real ou falso, o que encoraja o gerador a manter a variação dentro do lote, visto que se os exemplos forem muito similares, o discriminador saberá que é falso. O *feature mapping* modifica a função de custo do gerador para incluir a diversidade de lotes gerados, exigindo que o gerador gere dados que correspondam as estatísticas de dados reais. Esses métodos nos permitem treinar modelos que antes eram inutilizáveis.

Também com o mesmo intuito, Salimans et al. [23] propôs em 2016 as GANs Unrolled, em que é dada a oportunidade para o gerador desenrolar k passos no discriminador. Então o gerador é atualizado usando *backpropagation* com o custo calculado na etapa k final. Como demonstrado por Metz et al. [18], o uso dos valores da etapa k desestimula o gerador a explorar o local ótimo que neutraliza o discriminador. As desvantagens dessa abordagem são o aumento no tempo de treinamento, já que cada atualização do gerador simula atualizações de vários discriminadores e um cálculo de gradiente mais complicado [18].

Em 2017, Isola et al. [13] propôs uma GAN que utilizava uma imagem como entrada do gerador ao invés de ruído. Essa rede, chamada Pix2Pix, foi projetada para realizar o processo de tradução de imagem para imagem¹, tendo o cGAN como base. O modelo é treinado em pares de imagens (com resolução 256 x 256) e realiza a previsão de uma imagem de saída correspondente para qualquer imagem de entrada. O objetivo da Pix2Pix é produzir imagens realistas (para enganar o discriminador) e gerar imagens próximas da verdade básica, portanto, a perda de conteúdo é usada junto com a perda adversária na função objetivo, sendo que a perda de conteúdo mede a distância $L1$ [8] entre as duas imagens, a saída da rede e a imagem real [13]. Assim como as outras GANs, a Pix2Pix segue a mesma metodologia de treinamento proposta por Goodfellow et al. [10], conforme apresentado na Figura 2.6. Wu and Hall [28] menciona como esta rede se mostrou eficaz para uma variedade de tarefas de tradução de imagem para imagem, como por exemplo, placas de rua, rótulos de fachadas, bordas de fotos. Algumas destas tarefas e seus resultados são apresentadas na Figura 2.7.

2.2 Experimento e aquisição sísmica

Um experimento sísmico consiste em realizar procedimentos na superfície terrestre para descobrir o que está nas sub-superfícies. O objetivo é descobrir o que há no interior sem realizar perfuração, sendo a propagação de ondas crucial para isto. Neste contexto, a onda sísmica é gerada pelo homem para fazer a análise das sub-superfícies, onde o tempo de propagação diz onde estão e a amplitude ajuda na descoberta do que há nessa sub-superfície. Durante o experimento, uma fonte controlada (S) de onda sísmica é usada repetidamente, sendo cada acionamento ou disparo dessa fonte chamado de tiro. Na etapa de aquisição, vários tiros são disparados e seus resultados são registrados em um conjunto de receptores (G) [12]. Cada G realiza o registro temporal da pressão ou movimento das partículas ao seu redor, tendo uma determinada taxa de amostragem, comumente 4 ms em se tratando da sísmica de reflexão [4], conforme apresentado na Figura 2.8.

Sísmica de reflexão é o método mais comum de experimentação sísmica, onde as ondas

¹A tradução é processo de mapeamento de uma imagem de entrada para uma imagem de saída [13].

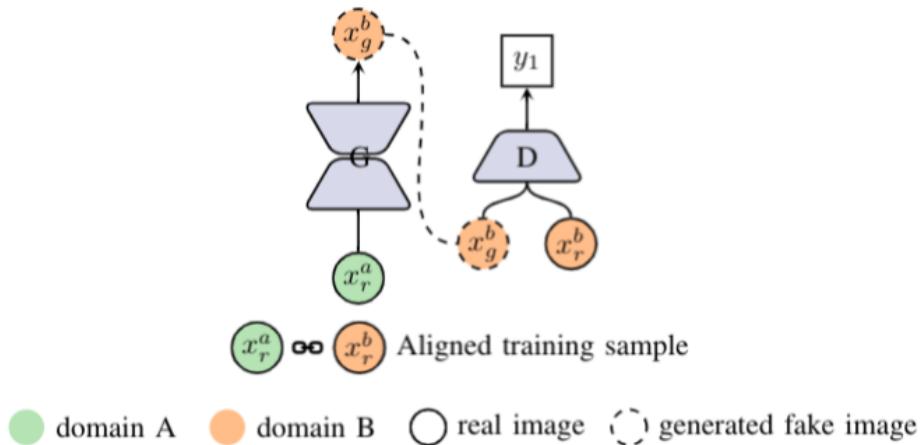


Figura 2.6: Processo de treinamento da Pix2Pix: primeiro, G recebe uma imagem real do domínio A (x_r^a), gerando uma falsa do domínio B (x_g^b). Em seguida a falsa é passada para o D juntamente com a real do domínio B (x_r^b) que analisa o par entrada-alvo e realiza a classificação em real e falsa. Figura adaptada de Yi et al. [29]).

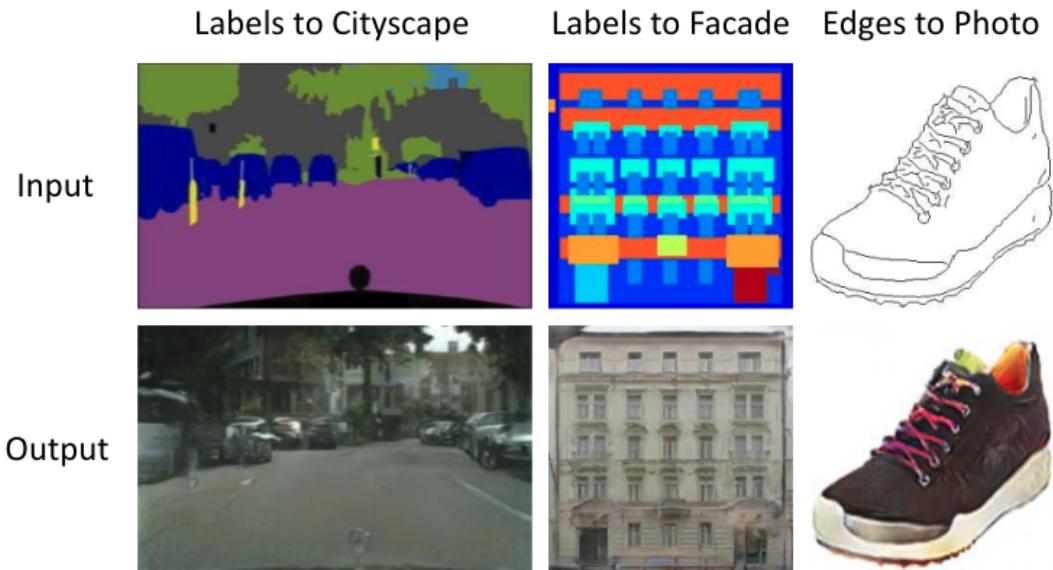


Figura 2.7: Exemplos de tarefas realizada pela Pix2Pix e os resultados obtidos. Figura reproduzida de Isola et al. [13].

são introduzidas e ao atingir uma **interface** entre regiões com diferentes densidades parte da energia é transmitida através da interface e outra parte é refletida. Na superfície, o registro pelos receptores em função do tempo do sinal que refletido é chamado de traço sísmico. Sendo este a combinação da resposta do ambiente com o pulso de origem usado [12]

Um traço está associado a somente um par (S, G), ou seja, para uma fonte e um receptor há apenas um traço que representa as amplitudes recebidas por um receptor

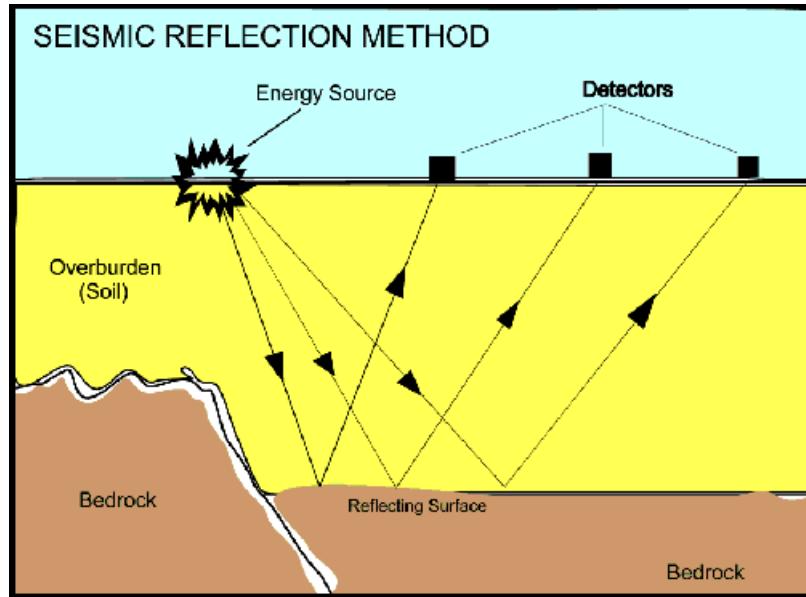


Figura 2.8: O tiro disparado pela fonte (*Energy Source*) causa a propagação das ondas até uma subsuperfície refletora (*Reflecting Surface*) que são refletidas e capturadas pelos receptores (*Detectors*). Figura reproduzida de Resources [21]

(G) ao longo do tempo **provindo** de uma fonte (S). Após este processo, esses dados são armazenados em arquivos digitais, tais como o SEG-Y (exemplo do dado sísmico real é apresentado na Figura 2.9), onde podem ser processados, migrados e interpretados. Na etapa de interpretação, especialistas analisam os dados a fim de descobrir o conteúdo das subsuperfícies, sendo o rastreamento de horizontes um dos processos realizados do decorrer dessa análise.

2.3 Rastreamento de horizontes

Os horizontes sísmicos são utilizados para construir modelos de estrutura e estratigrafia de superfícies geologicamente importantes. O seu rastreamento é um problema desafiador e que consome muito tempo do intérprete, sendo, a diminuição do tempo humano nesse processo tema de pesquisa ativo na sísmica [27].

O rastreamento de horizonte é um passo importante para a interpretação sísmica [27], sendo este realizado por um intérprete, responsável por escolher a técnica mais eficiente e que permitirá a obtenção da melhor interpretação [7]. Na Figura 2.10 é mostrado um exemplo de rastreamento de horizontes em um dado sísmico volumétrico, em que as curvas em verde marcam um horizonte sísmico de interesse.

As técnicas de rastreamento de horizontes podem ser ordenadas em *picking* manual, interpolação e *autopicking*, ao considerarmos suas eficiências interpretativas do menos eficiente ao mais eficiente [7]. No rastreamento *picking* manual os horizontes são interpretados em linhas, linhas cruzadas, intervalos de tempo e percursos em que o intérprete procura por continuidade e similaridade local nos dados a fim de identificar o evento a ser escollido [7]. Nesta abordagem é o intérprete que realiza todo o processo, sendo, portanto, o

² Voxel é a extensão do conceito de pixel para três dimensões.

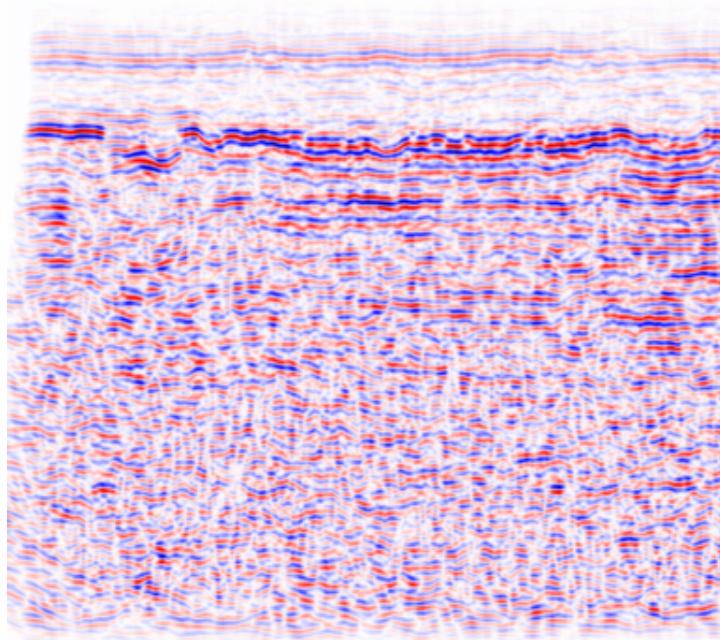


Figura 2.9: Dados reais adquiridos na bacia brasileira de Solimão. O eixo vertical representa o tempo, o eixo horizontal representa o *Common Midpoint* (CMP) e as cores representam os valores das frequências, variando de -1 (azul) a +1 (vermelho).

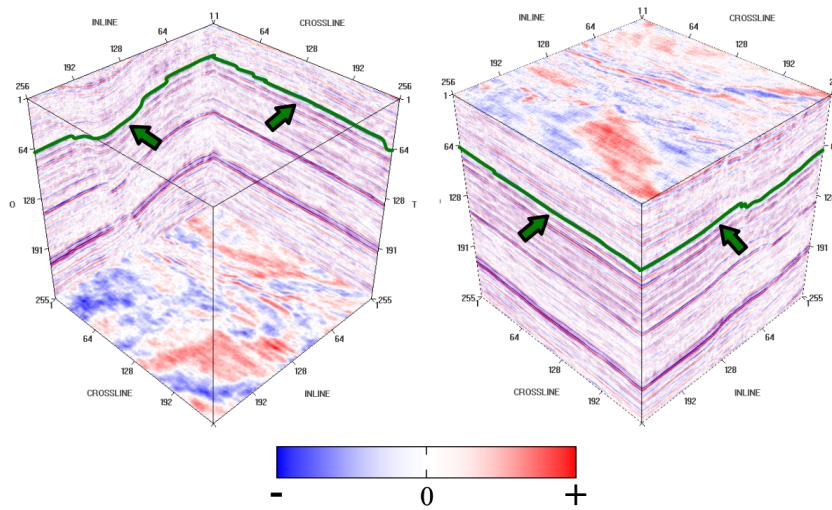


Figura 2.10: Duas vistas de um dado sísmico volumétrico onde as cores representam os valores de amplitude de cada *voxel*². As curvas em verde, apontadas pelas setas, marcam um horizonte sísmico de interesse no dado. Figura reproduzida de Carvalho [6].

caminho menos eficiente [7, 27], ao ser considerado o tempo necessário para executá-lo, por exemplo. Já a interpolação é mais eficiente que a anterior, mas depende do horizonte ser localmente suave. Nesse caso, se essa característica não for verdadeira entre os pontos de controle (por exemplo, há uma falha entre as linhas interpretadas), os resultados serão imprecisos [7]. Por fim, o *autopicking* já utilizado para apoiar a interpretação desde o início dos anos 80, consiste em sistemas interativos onde o interprete escolhe alguns

pontos como sementes **em linhas ou linhas cruzadas no levantamento 3D**. Por sua vez, esses pontos são usados como controle inicial para o rastreamento, no qual o algoritmo procura características semelhantes em seus vizinhos. Se tal recurso for encontrado dentro das restrições especificadas, ele seleciona esse traço e avança para o próximo [27]. Na Figura 2.11 é mostrado um esboço da execução deste processo, onde o *autotracker* interromperá o rastreamento se qualquer um dos critérios de pesquisa não for atendido.

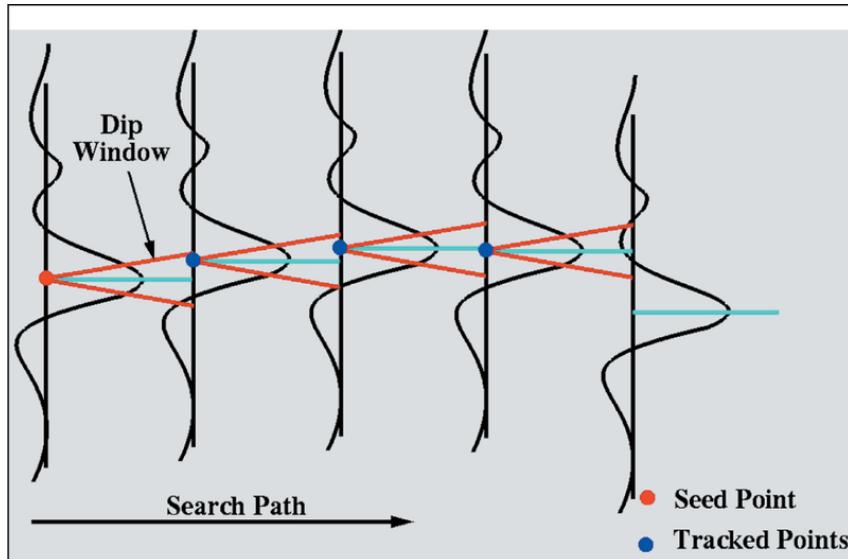


Figura 2.11: Esboço da operação de *picking* automático. O ponto vermelho (*Seed Point*) representa a semente fornecida no primeiro traço, e os pontos azuis (*Tracked Points*) são os rastreados pelo ferramenta de *tracking* automática. Neste caso, foi determinada um janela (*Dip Window*) e uma amplitude que foram atendidas até o penúltimo traço. Figura reproduzida de Dorn [7].

Capítulo 3

Trabalhos relacionados

Esse capítulo apresenta os trabalho relacionados ao nosso trabalho. A Subseção 3.1 apresenta algumas propostas de automatização do rastreamento de horizontes que foram realizadas nas últimas décadas, enquanto a Seção 3.2 apresenta os trabalhos da literatura em que as GANs foram aplicada para a otimização de dados sísmicos.

3.1 Horizon Tracking

Diversos métodos baseados em interpolação têm sido propostos para realizar o rastreamento de horizontes, como por exemplo, Zeng et al. [30] propuseram um método no qual deveriam ser inicialmente selecionados vários horizontes de referência e com isso era interpolado o volume de horizonte. Entretanto, Wu and Zhang [27] afirma, que há a necessidade de os horizontes sísmicos serem suaves ou mesmo lineares entre pontos de controle, entretanto os horizontes são imprecisos perto de falhas e outras zonas geológicas complicadas.

Alguns métodos automáticos também foram propostos, como por Wu and Zhang [27], que realiza a segmentação semântica para rastrear horizontes usando DNNs com a arquitetura de *encoder-decoder* proposta por Long et al. [15]. Para isso, foi realizado manualmente o picking de alguns horizontes para construir os horizontes de referência, sendo parte destes dados utilizados no treinamento e outras no teste. Para cada segmento de traços sísmicos que são limitados nos mesmos horizontes foram aplicados os mesmos rótulos (cores), e com esses traços rotulados foi realizado o treinamento da rede, onde a saída consiste de traços sísmicos segmentados, em que o limite de cada segmento representa os horizontes. No entanto, esse processo deve ser refeito para cada novo volume de dados sísmicos e, quando os dados sísmicos são imprecisos, o processo de segmentação manual pode se tornar impossível de realizar.

3.2 GANs

Embora as GANs ainda não tenham sido utilizadas para melhorar o rastreamento de horizontes, alguns trabalhos as utilizaram para prover melhorias em dados sísmicos. Entre eles, o trabalho desenvolvido por Richardson [22] treinou uma DCGAN para produzir

modelos de velocidade realistas, onde foi demonstrado que o gerador pode aprender a produzir modelos de velocidade plausíveis a partir de vetores latentes. Como exemplo, o método foi usado para redução da ordem de modelo em *Full-Waveform Inversion* [9].

Siahkoohi et al. [25] propôs um novo método baseado em GANs para a reconstrução rápida e precisa de dados sísmicos altamente sub-amostrados (independentemente do tipo de amostragem), visto que a aquisição de dados densamente amostrados é um dos maiores desafios na imagiologia sísmica. Com esse método eles foram capazes de reconstruir dados sísmicos, conforme apresentado na Figura 3.1. Já Lu et al. [16] utilizou uma GAN para melhorar a interpolação de falhas, na qual ela foi desenvolvida para aumentar a resolução dos dados sísmicos sem modificar a amplitude e a frequência do espectro útil do dado que foi utilizado em um rede neural que detecta faltas . **frequência**), conforme apresentado na Figura 3.2.

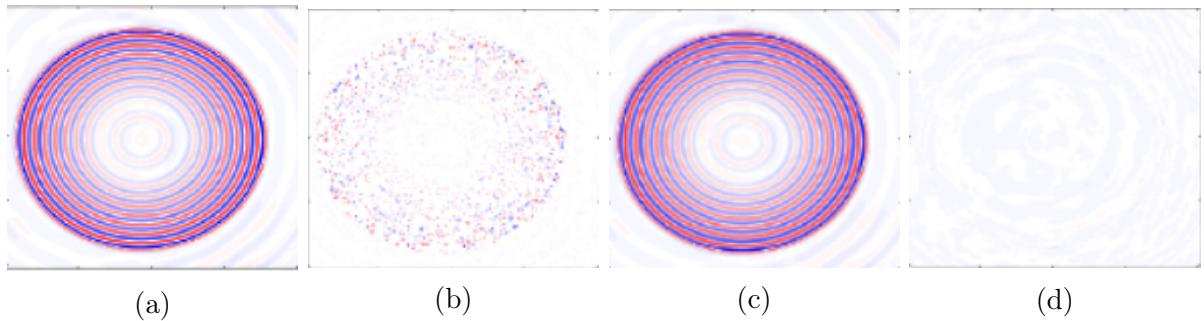


Figura 3.1: Em (a) temos o dado original com regiões removidas de forma aleatória gerando (b). Aplicando o gerador da GAN em (b) obtém-se o dado reconstruído (c). Por fim, em (d) temos a diferença entre o dado original e o dado gerado pela rede. Figura reproduzida de Siahkoohi et al. [25].

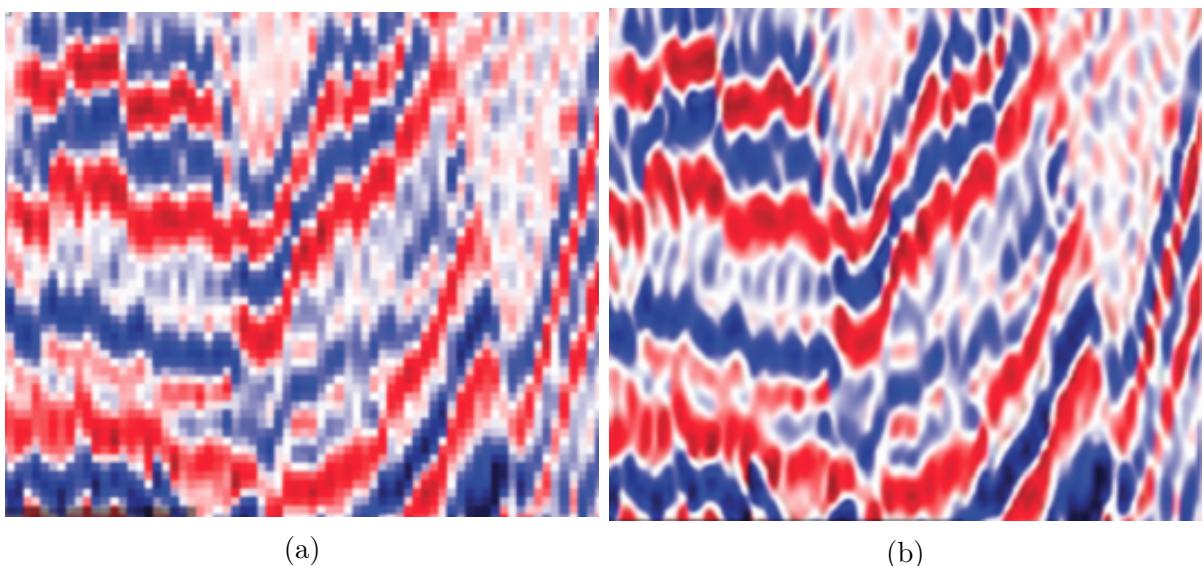


Figura 3.2: Em (a) temos o dado original e em (b) temos o dado gerado pela GAN. Figura reproduzida de Lu et al. [16].

Capítulo 4

Metodologia e Método

As GANs já foram aplicadas com sucesso aos dados sísmicos. Em Lu et al. [16] a rede apreende um filtro para gerar dados de alta resolução através da sua contrapartida de baixa resolução, mantendo as características principais dos dados originais. Uma característica similar é desejada no processo de remoção de artefatos: o dado de saída deve manter as estruturas do dados de entrada.

Os dados sísmicos serão trabalhados no domínio da imagem, reduzindo nosso problema para uma tradução de imagem para imagem, onde a imagem com a presença de artefatos será mapeada para a imagem sem os artefatos. A primeira GAN que será desenvolvida para realizar este processo será a Pix2Pix, apresentada na Seção 2.1.4. Selecionamos essa GAN para a realização do *baseline* devido aos seus bons resultados na tarefa de tradução de imagem para imagem em geral.

Para realizar o treinamento dessa rede é necessário um conjunto de dados rotulados (os rótulos nesse caso é a imagem alvo) e devido ausência de um procedimento para realizar a remoção de artefatos em dados reais e a impossibilidade de remove-los manualmente, utilizarem dados sintéticos. Uma especificação de como esses dados vão ser criados é apresentado na Seção 4.1.

Posteriormente, após a obtenção de resultados satisfatórios em dados sintéticos, utilizaremos dados reais, disponibilizados pelo Centro Estudo de Petróleo (CEPETRO), conforme fluxo apresentado na Figura 4.1. Durante o treinamento com dados sintéticos, a validação dos resultados podem ser realizado através da diferença elemento por elemento entre a dado gerada e a dado nítida. Entretanto, quando trabalhos com dados reais não temos a imagem nítida, e portanto a validação deste resultados não é trivial, pois não se temos um método numérico de verificar se a dado gerada é melhor do que a dado original. Com isso, inicialmente, a principal fonte de validação dos dados gerados a parti de dados reais será a analise destes dados por especialistas em sísmica do CEPETRO, e a utilização desse dado em uma das atividades práticos realizados por estes especialistas.

A remoção de artefatos poderá beneficiar diversas aplicações, entretanto, o rastreamento de horizontes foi selecionado para realizar a validação dos nossos resultados. A Figura 4.2 apresenta o novo fluxo desse processo, onde o dado sísmico será convertido para o domínio da imagem, passado pelo gerador que gerará uma imagem de saída que será novamente convertida para dado sísmico e, por fim, utilizado no rastreamento de horizontes.

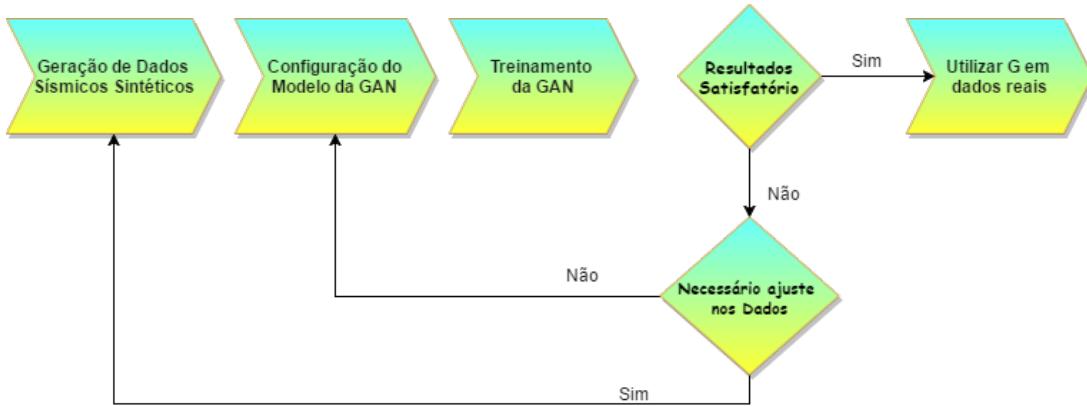


Figura 4.1: Fluxo do trabalho: inicialmente serão gerados dados sintéticos, onde a rede será treinada e aplicada em dados reais quando os resultados obtidos forem satisfatórios.

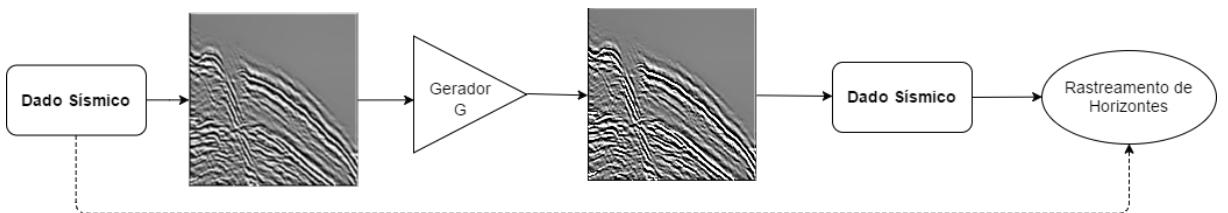


Figura 4.2: As setas contínuas representam o novo fluxo proposto para o rastreamento de horizontes, enquanto as tracejadas representam o fluxo convencional.

4.1 Dataset

Produziremos o *dataset* necessário para este projeto com o apoio de especialistas em sísmica. O primeiro passo consistirá em gerar as estruturas para modelar tanto os dados com artefatos quanto os nítidos. Os dados com artefatos devem representar os dados reais, tendo em vista que na aquisição também são capturadas interferências produzidas por fenômenos da natureza e da própria onda que foi introduzida, para isso, o modelaremos através da convolução das estruturas com uma PSF extraída de dados sísmicos reais de difrações. Para a produção da imagem nítida, as estruturas serão convolvidas com Ricker wavelet — modelo de aquisição perfeita. Com isso, teremos o par de imagens necessárias para que a imagem com artefatos seja mapeada para uma imagem nítida.

4.2 Cronograma

A Tabela 4.1 apresenta o cronograma dividido por trimestres, contendo as atividades a serem concluídas e o tempo esperado para cada uma delas.

Atividades	1 nd ano				2 nd ano			
	1	2	3	4	1	2	3	4
Cumprir os requisitos do curso	x	x	x	x				
Revisão da literatura			x	x	x	x	x	x
Elaborar a proposta de dissertação			x	x	x			
Exame de qualificação de mestrado					x			
Experimentos					x	x	x	
Publicação dos resultados da pesquisa					x	x	x	x
Defesa da dissertação de mestrado								x

Tabela 4.1: Cronograma de atividades

Capítulo 5

Experimentos e Resultados

Na Seção 5.1 serão apresentados alguns experimentos realizados com diferentes arquiteturas e modelos de GANs. Enquanto na Seção 5.2, apresentamos os experimentos quanto a criação de dados sísmicos sintéticos e a utilização das GANs para aprender a PSF reversa.

5.1 GANs

Nossos primeiros experimentos consistiram no estudo da primeira GAN proposta para a criação de imagens sintéticas, a DCGAN. Fizemos nossa própria implementação dessa arquitetura e realizamos o treinamento nos *datasets* MNIST¹ e CIFAR10² utilizando a biblioteca GANLib³ que estamos desenvolvendo. Um exemplo de dado sintético gerado é mostrado na Figura 5.1.

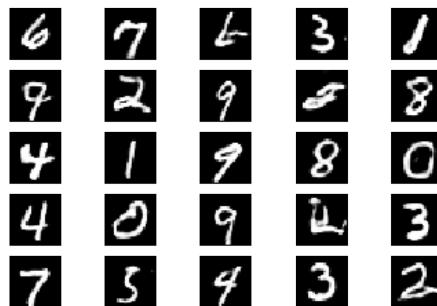


Figura 5.1: Amostras do *dataset* MNIST geradas com a DCGAN.

Nos experimentos seguintes, reproduzimos os experimentos realizados no artigo da Pix2Pix [13]. Os resultados obtidos foram equivalentes aos apresentados no artigo. Em seguida, fizemos as adaptações necessárias para treinar o nosso próprio *dataset*. Geramos um *dataset* de teste com curvas aleatórias coloridas para serem interpoladas. A imagem de entrada consiste de curvas tracejadas e a imagem objeto consiste das mesmas curvas continuas. Após o treinamento da Pix2Pix, conseguimos realizar as interpolações, conforme apresentados na Figura 5.2.

¹<http://yann.lecun.com/exdb/mnist/>

²<https://www.cs.toronto.edu/~kriz/cifar.html>

³<http://bit.ly/JonlenesGANLib>

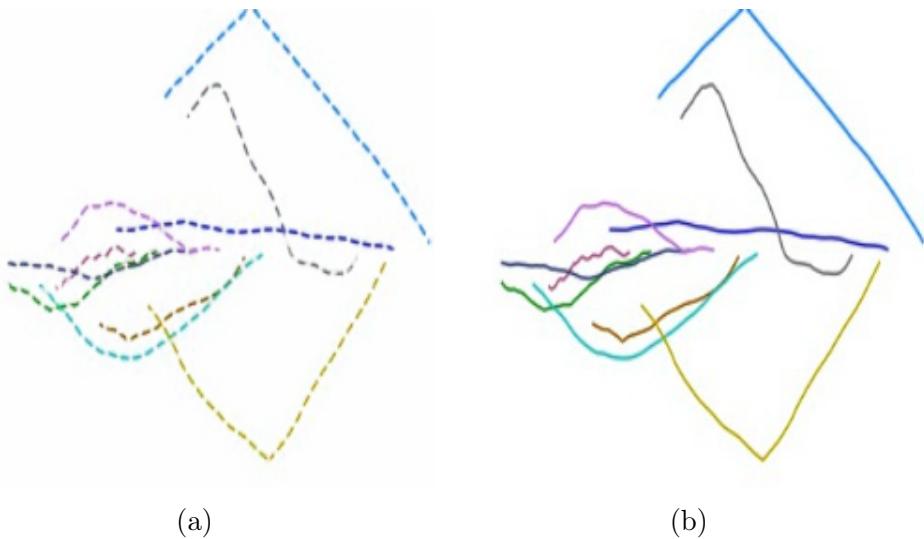


Figura 5.2: Dada a imagem (a) de entrada a rede gera a imagem (b) de saída.

5.2 Sísmica

No primeiro experimento, as estruturas foram geradas de duas formas: a primeira utilizou algoritmos de detecção de arestas em dados sísmicos reais, seguidos de pós-processamento necessários para o ajuste das estruturas (Figura 5.3a esquerda) e cria-las de modo pseudoaleatório (Figura 5.3a direita). O segundo passo consistiu na realização da convolução dessas estruturas com a PSF para gerar dados com artefatos. Foram extraídas 6 PSFs de dados reais de difração, sendo escolhido uma delas aleatoriamente na convolução com as estruturas. Depois de selecionar uma das PSF, calcula-se sua frequência para ser utilizada na Ricker. Finalmente, para gerar os dados sem artefatos, as estruturas foram convolvidas com Ricker (na Figura 5.3b direita, mostramos um exemplo correspondente a PSF da Figura 5.3b). Na Figura 5.4 é apresentada a evolução do experimento quanto a geração de estruturas pseudoaleatórios.

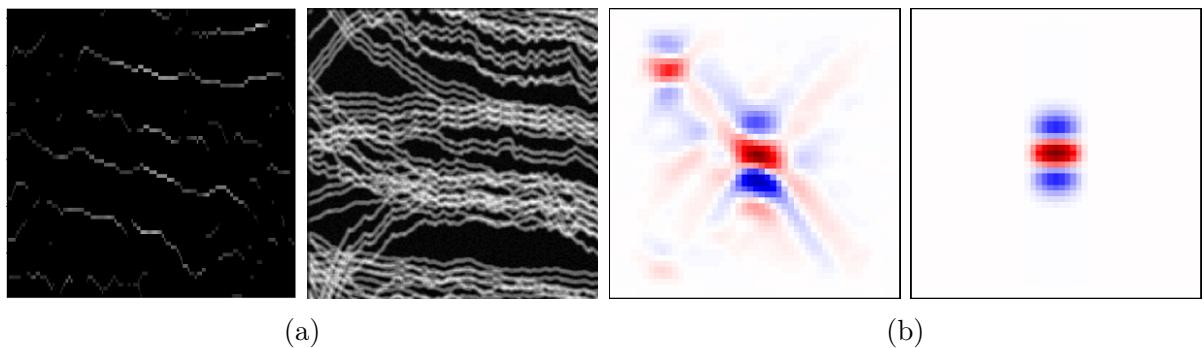


Figura 5.3: Em (a) são apresentados dois exemplos de estruturas (tamanho 256x256 pixel): extraída de dados sísmicos reais (esquerda) e gerada de forma pseudoaleatório (direita), enquanto em (b) é mostrada uma das PSFs (esquerda) com tamanho 41x41 pixel e Ricker (direita) redimensionada.

Utilizando esses métodos, geramos 10 mil pares de imagens para o treinamento, 2 mil para validação e 2 mil para teste de acordo com os protocolos recomendados para a

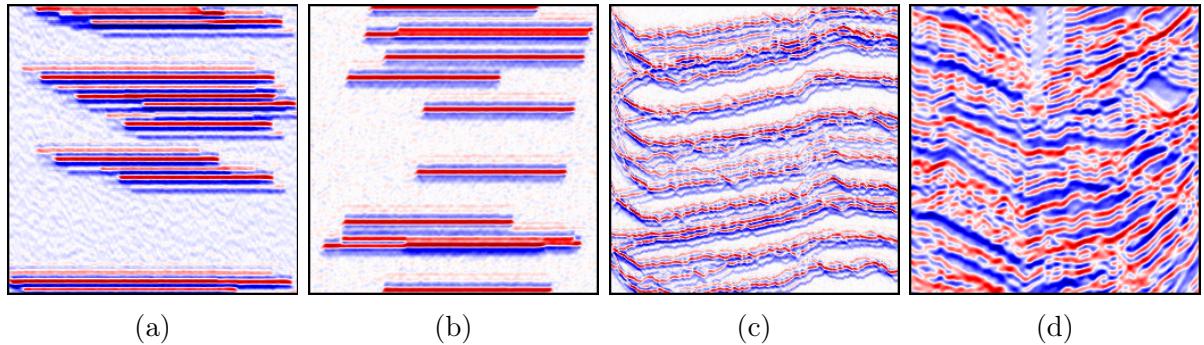


Figura 5.4: Evolução dos dados sintéticos com estruturas pseudo-aleatórias: (a) inicialmente geramos apenas algumas estruturas lineares aleatórias. Depois disso, fizemos algumas melhorias e geramos uma segunda versão (b). Após treinar a rede com as retas, geramos estruturas não lineares (c). Novamente realizamos melhorias e geramos a versão final do dado deste experimento (d).

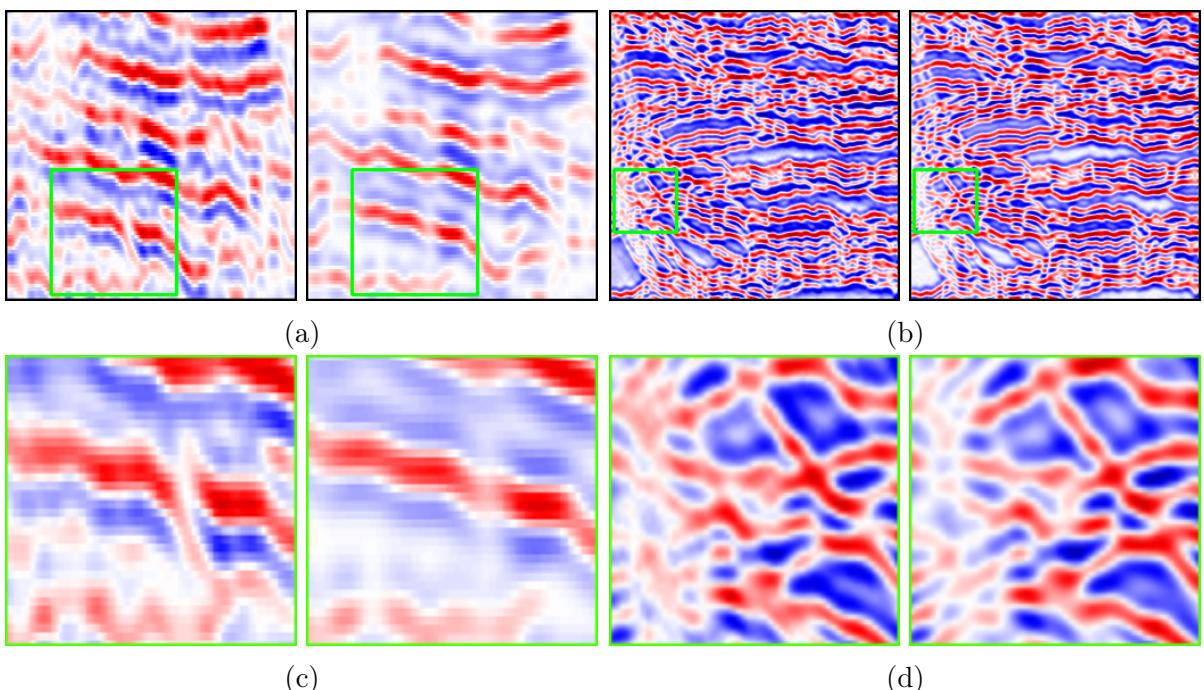


Figura 5.5: Amostra sintética: o eixo vertical representa o tempo, o eixo horizontal representa o *Common Midpoint* (CMP) e as cores representam os valores das frequências, variando de -1 (azul) a +1 (vermelho). Na Figura (a) temos a imagem com artefatos (esquerda) e a imagem nítida (direita) geradas com as estruturas extraídas de dados reais, enquanto na Figura (b) temos uma amostra gerada com as estruturas pseudo-aleatórias. Finalmente, nas Figuras (c) e (d) temos uma ampliação da região destacada das Figuras (a) e (b), respectivamente.

execução de experimentos em Machine Learning — área macro de *Deep Learning*. Todas as imagens têm tamanho 256 x 256 para serem usadas na Pix2Pix. Com este conjunto de dados, a rede foi treinada por 800 épocas (aproximadamente 63 mil iterações) por 24 horas em uma GPU Nvidia Tesla v100. O resultado pode ser visto na Figura 5.6. Aplicamos essa rede em dados reais adquiridos na bacia brasileira do Solimões e o resultado em uma

amostra de teste é apresentado na Figura 5.7.

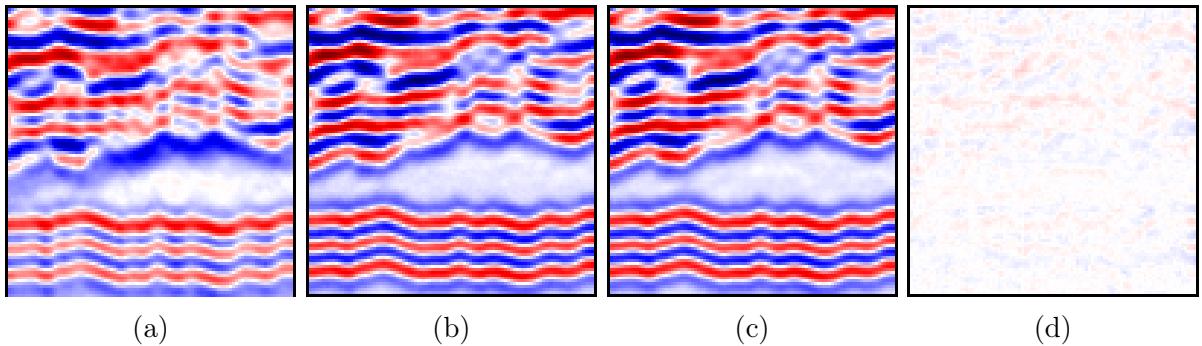


Figura 5.6: A primeira Figura (a) é uma imagem com artefatos, já em (b) temos a imagem nítida correspondente. Enquanto isso, em (c) apresentamos a imagem nítida gerada pela nossa rede e (d) mostramos a diferença entre as imagens nítidas reais e geradas pela rede. Isso mostrar que a rede conseguiu aprender a PSF reversa, neste caso.

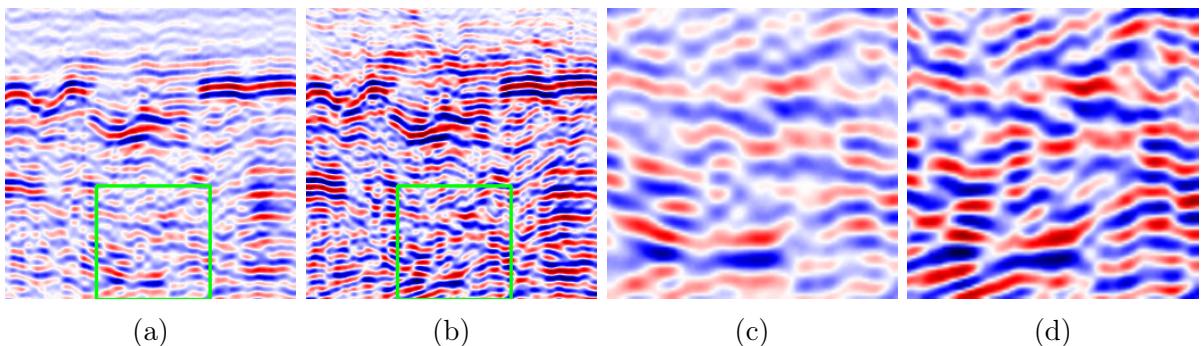


Figura 5.7: Dados reais adquiridos na bacia brasileira do Solimão. A Figura (a) é uma amostra recortada do dado original e a Figura (b) é a imagem gerada pela nossa rede para essa amostra. Finalmente, nas Figuras (c) e (d) temos uma ampliação da região destacada das Figuras (a) e (b), respectivamente.

Os dados sísmicos sintéticos destes experimentos ainda não representem adequadamente os dados reais, entretanto, a rede aprendeu a estimar a PSF reversa. Nos próximos experimentos exploraremos a geração das estruturas e também adicionaremos ruído na imagem com artefatos (branco, rosa, extraídos de dados reais, etc.) com o intuito de obtermos dados representativos dos dados reais no treinamento e prover as melhorias esperadas em dados reais. Por fim, também validaremos o método através da preservação dos coeficientes de reflexão.

Capítulo 6

Conclusão

Este documento apresentou uma proposta para o exame de qualificação de mestrado. O projeto abordará o tópico de remoção de artefatos e rastreamento de horizontes, temas de pesquisa ativos na sísmica. Especificamente, propomos uma técnica que emprega uma GAN para remover artefatos de dados sísmicos em tempo. Para isso usamos dados sintéticos para treinar a rede e, posteriormente, aplicamos aos dados reais. Uma vez que o treinamento da rede esteja completo, o processo de remoção de artefatos pode ser realizado rapidamente, processo este que visa melhorar a qualidade dos dados sísmicos reais e beneficiar diversas aplicações. As atividades foram planejadas para serem concluídas em dois anos, e como resultado espera-se um modelo robusto para a remoção de artefatos em dados sísmicos.

Referências Bibliográficas

- [1] Martin Arjovsky and Léon Bottou. Towards Principled Methods for Training Generative Adversarial Networks. *arXiv preprint arXiv:1701.04862*, pages 1–17, 2017. ISSN 0021-9193. doi: 10.2507/daaam.scibook.2010.27. URL <http://arxiv.org/abs/1701.04862>.
- [2] Duhyeon Bang and Hyunjung Shim. MGGAN: Solving Mode Collapse using Manifold Guided Training. *arXiv preprint arXiv:1804.04391*, 2018. ISSN 0300-9629. doi: 10.1016/0300-9629(79)90638-8. URL <http://arxiv.org/abs/1804.04391>.
- [3] Mike Bernico. *Deep Learning Quick Reference*. Packt Publishing, 2018. ISBN 9781788837996.
- [4] Ricardo Biloti. Processamento sísmico, 2018.
- [5] Owen Carey. Generative adversarial networks (gans). <https://towardsdatascience.com/generative-adversarial-networks-gans>, 2013.
- [6] Paulo Cezar Carvalho. *Visualização Volumétrica de Horizontes em Dados Sísmicos 3D*. PhD thesis, PUC-Rio, 2004.
- [7] Geoffrey Dorn. Modern 3-d seismic interpretation. In *Modern 3-D seismic interpretation*, volume 17, pages 1262–1272. The Leading Edge, 1998.
- [8] Pedro F Felzenszwalb and Daniel P Huttenlocher. Distance transforms of sampled functions. *Theory of computing*, 8(1):415–428, 2012.
- [9] EOS Seismic Laboratory for Imaging and Modeling. Full-waveform inversion, 2015.
- [10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.
- [11] J.W.J. Hosken. Ricker wavelets in their various guises. *First Break*, 6, 01 1988. doi: 10.3997/1365-2397.1988002.
- [12] Luc T Ikelle and Lasse Amundsen. *Introduction to petroleum seismology*. Society of Exploration Geophysicists, 2018.

- [13] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, July 2017. doi: 10.1109/CVPR.2017.632.
- [14] Sergey Fomel Jon Claerbout Jesse Lomask, Antoine Guittot and Alejandro A. Valenciano. Flattening without picking: *Geophysics*, 71. P13–P20, 2006.
- [15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [16] Ping Lu, Matt Morris, Seth Brazell, Cody Comiskey, and Yuan Xiao. Using generative adversarial networks to improve deep-learning fault interpretation networks. *The Leading Edge*, 37(8):578, 2018. doi: 10.1190/tle37080578.1. URL <http://dx.doi.org/10.1190/tle37080578.1>.
- [17] Eric Maskin. Nash equilibrium and welfare optimality. *The Review of Economic Studies*, 66(1):23–38, 1999.
- [18] Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled Generative Adversarial Networks. *arXiv preprint arXiv:1611.02163*, pages 1–25, 2016. ISSN 10636919. doi: 10.1109/MSP.2017.2696576. URL <http://arxiv.org/abs/1611.02163>.
- [19] Simon Osindero. Conditional Generative Adversarial Nets. *arXiv preprint arXiv:1411.1784v1*, pages 1–7, 2014.
- [20] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, pages 1–16, 2016.
- [21] Geologic Resources. Seismic reflection description, 2018.
- [22] Alan Richardson. Generative adversarial networks for model order reduction in seismic full-waveform inversion, 2018.
- [23] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016.
- [24] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016.
- [25] Ali Siahkoohi, Rajiv Kumar, and F Herrmann. Seismic data reconstruction with generative adversarial networks. In *80th EAGE Conference and Exhibition 2018*, 2018.

- [26] Tracy J. Stark. Unwrapping instantaneous phase to generate a relative geologic time volume. In *SEG Technical Program Expanded Abstracts 2003*, pages 1707–1710. Society of Exploration Geophysicists, 2003.
- [27] Hao Wu and Bo Zhang. A deep convolutional encoder-decoder neural network in assisting seismic horizon tracking. *arXiv preprint arXiv:1804.06814*, 2018.
- [28] Kun Xu Wu, Xian and Peter Hall. A survey of image synthesis and editing with generative adversarial networks. *Tsinghua Science and Technology*, 22(6):660–674, December 2017. ISSN 1007-0214. doi: 10.23919/TST.2017.8195348.
- [29] Xin Yi, Ekta Walia, and Paul Babyn. Generative adversarial network in medical imaging: A review. *arXiv preprint arXiv:1809.07294*, 2018.
- [30] Hongliu Zeng, Milo M. Backus, Kenneth T. Barrow, and Noel Tyler. Stratal slicing, part i: realistic 3-d seismic model. *Geophysics*, 63(2):502–513, 1998.
- [31] Yuxuan Zhang. Image generation with dcgan. http://www.timzhangyuxuan.com/project_dcgan, 11 2016.