CEFET/RJ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO

Análise de Arquiteturas de Redes Neurais Para Segmentação Semântica de Dados Sísmicos

Gabriel Danilo Figueiredo da Silva

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, como parte dos requisitos necessários à obtenção do grau de Mestre em Engenharia Elétrica.

Orientadores:
João Terêncio Dias
Luciana Faletti Almeida

Ficha catalográfica elaborada pela Biblioteca Central do CEFET/RJ

S586 Silva, Gabriel Danilo Figueiredo da
Análise de arquiteturas de redes neurais para segmentação semântica de dados sísmicos / Gabriel Danilo Figueiredo da Silva. — 2023.
70f. : il. color. , enc.

Dissertação (Mestrado) Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, 2023.
Bibliografia : f. 65-70
Orientador: João Terêncio Dias
Coorientadora: Luciana Faletti Almeida

1. Redes neurais (Computação). 2. Arquitetura de redes de computador. 3. Banco de dados. 4. Petróleo e gás. 5. Geologia. I. Dias, João Terêncio. (Orient.). II. Almeida, Luciana Faletti. (Cooriet.). III. Ttulo.

CDD 006.3

Elaborada pela bibliotecária Tania Mello – CRB/7 nº 5507/04

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA

GABRIEL DANILO FIGUEIREDO DA SILVA

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, como parte dos requisitos necessários à obtenção do grau de Mestre em Engenharia Elétrica.

Documento assinado digitalmente
gov.br JOAO TERENCIO DIAS
Data: 31/05/2023 16:23:55-0300
Verifique em https://validar.iti.gov.br

Orientador: Prof. Dr. João Terêncio Dias.
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca - CEFET/RJ

Documento assinado digitalmente
gov.br LUCIANA FALETTI ALMEIDA
Data: 31/05/2023 18:43:34-0300
Verifique em https://validar.iti.gov.br

Co-Orientador:Prof. Dra. Luciana Faletti Almeida.
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca - CEFET/RJ.

Documento assinado digitalmente
gov.br MILENA FARIA PINTO
Data: 31/05/2023 19:56:31-0300
Verifique em https://validar.iti.gov.br

Prof. Dra. Milena Faria Pinto.
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca - CEFET/RJ.

Documento assinado digitalmente
gov.br KARLA TEREZA FIGUEIREDO LEITE
Data: 03/06/2023 12:45:18-0300
Verifique em https://validar.iti.gov.br

Prof. Dra. Karla Tereza Figueiredo Leite.
Universidade do Estado do Rio de Janeiro - UERJ.

Documento assinado digitalmente
gov.br GABRIEL MATOS ARAUJO
Data: 01/06/2023 17:01:28-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Gabriel Matos Araújo.
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca - CEFET/RJ

Documento assinado digitalmente
gov.br EUGENIO DA SILVA
Data: 04/06/2023 18:21:35-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Eugênio da Silva.
Universidade do Estado do Rio de Janeiro - UERJ-ZO.

Rio de Janeiro, Maio de 2023.

Resumo

Com os avanços tecnológicos, a caracterização de reservatórios de petróleo se tornou mais precisa e complexa, especialmente com a utilização de dados sísmicos tridimensionais (3D) para a localização de reservatórios de petróleo e gás. Nesta dissertação, são apresentadas duas arquiteturas de redes neurais profundas aplicadas à segmentação semântica de imagens sísmicas em 2 bancos de dados públicos, o banco de dados Holanda F3 e Penobscot. O treinamento da rede é realizado utilizando apenas 100 imagens para simular um cenário onde há pouca quantidade de dados disponíveis. A proposta é utilizar uma arquitetura convolucional amplamente conhecida, chamada U-Net junto a um filtro de Equalização de Histograma Adaptativo Limitada por Contraste (*Contrast Limited Adaptive Histogram Equalization* - CLAHE) na arquitetura, com o propósito de melhorar a nitidez da imagem e assim, melhorando seu desempenho. Essa arquitetura foi comparada com outra arquitetura chamada Danet-FCN3 que é uma rede bem conhecida no campo de segmentação de imagens sísmicas. Entretanto, um dos pontos fracos dessa arquitetura é sua alta complexidade computacional. Por isso, foi realizada uma pesquisa para manter o alto desempenho, reduzindo a complexidade da rede, e o tempo de treino. Os resultados mostram uma melhoraria de até 10 % nos testes realizados em ambos bancos de dados, quando comparado à sua versão sem o filtro CLAHE e também conseguiu atingir 99,37% de acurácia, superando assim o estado-da-arte obtido pela arquitetura Danet-FCN3, que atingiu uma precisão de 98,13%. Outro ponto estudado durante esse trabalho foi como a variação do tamanho dos *tiles* impactam o desempenho final das redes. Com a métrica de mIoU (*Mean Intersection Over Union*) foi alcançado $90,02\%$ de desempenho. Esse resultado se aproxima do estado-da-arte obtido pela arquitetura Danet-FCN3 para essa métrica, que atingiu uma acurácia de $91,41\%$, sendo que nossa proposta usa uma abordagem substancialmente menos complexa.

Palavras-chaves: U-Net, Danet-FCN3, Segmentação Semântica.

Abstract

With technological advancements, the characterization of oil reservoirs has become more precise and complex, especially with the use of three-dimensional (3D) seismic data for the location of oil and gas reservoirs. In this dissertation, two deep neural network architectures applied to the semantic segmentation of seismic images are presented using two public databases, the Holanda F3 and Penobscot databases. The network training is performed using only 100 images to simulate a scenario with limited data availability. The proposal is to use a well-known convolutional architecture called U-Net along with a Contrast Limited Adaptive Histogram Equalization (CLAHE) filter in the architecture to improve image sharpness and, consequently, enhance its performance. This architecture is compared with another architecture called Danet-FCN3, which is well-known in the field of seismic image segmentation. However, one of its weaknesses is its high computational complexity. Therefore, research was conducted to maintain high performance while reducing the network's complexity and training time. The results show an improvement of up to 10% in the tests conducted on both databases when compared to the version without the CLAHE filter. It also achieved an accuracy of 99.37%, surpassing the state-of-the-art obtained by the Danet-FCN3 architecture, which achieved an accuracy of 98.13%. Another point studied during this work was how the variation in the size of the tiles impact the final performance of the networks. With the mIoU (Mean Intersection Over Union) metric, 90.02% of performance was achieved. This result is close to the state-of-the-art obtained by the Danet-FCN3 architecture for this metric, which reached an accuracy of 91.41%, and our proposal uses a substantially less complex approach.

Keywords: U-Net, Danet-FCN3, Semantic Segmentation.

Lista de ilustrações

Figura 1 – Descrição do cálculo do mIoU 16
Figura 2 – Geração de ondas sísmicas [1]. 18
Figura 3 – Leitura de aquisição em terra. Rn: número do receptor; em (b) traços sísmicos simplificados correspondentes às interfaces em (a) [2]. 18
Figura 4 – Esquema de aquisição sísmica em terra [3]. 19
Figura 5 – Aquisição sísmica no mar [4]. 19
Figura 6 – A imagem (A) mostra um dado sísmico cru. As linhas coloridas em (B) representam as geometrias identificadas na seção sísmica [5]. 20
Figura 7 – Diferença entre *Inline* e *Crossline* no banco de dados da Holanda F3 [6]. 21
Figura 8 – Exemplo de algumas seções sísmicas que compõem o volume sísmico do banco de dados da Holanda F3 [7]. 21
Figura 9 – Exemplo de uma imagem sísmica "gather" [8]. 22
Figura 10 – NMO e *stacking* [8]. 23
Figura 11 – técnicas de processamento "gather" [9]. 24
Figura 12 – (A) Domo de sal [10] e (B) Canais sísmicos (na cor azul) [11]. 25
Figura 13 – Horizontes com falhas [12]. 26
Figura 14 – Nove horizontes interpretados mostrados junto com duas seções sísmicas. [13]. 27
Figura 15 – Sete horizontes interpretados mostrados junto com duas seções sísmicas. [14]. 29
Figura 16 – As subdivisões da área de inteligência artificial. 31
Figura 17 – Exemplo de convolução 2D. 33
Figura 18 – Representação simplificada de um conjunto comum de camadas presentes em uma CNN. 34
Figura 19 – Exemplo de funções de ativações, em (A) a função ReLU, em (B) a função Sigmoide e em (C) a Tangente Hiperbólica 35
Figura 20 – Exemplo de um novo mapa de características a partir de um filtro *max pooling* de 2x2 com $stride = 2$. 35
Figura 21 – A arquitetura Alexnet. 37
Figura 22 – Diagrama em blocos de um sistema utilizando aprendizado supervisionado. 37
Figura 23 – Classificador de 2 classes. 38
Figura 24 – Descrição da fase de propagação. 40
Figura 25 – Descrição da fase de retropropagação. 41
Figura 26 – (a) Unidade residual com atalho de identidade. (b) Unidade residual com atalho de projeção 43
Figura 27 – Segmentação semântica [15]. 44

Figura 28 – Arquitetura de uma rede *Encoder − Decoder* [16]. 46
Figura 29 – (1) - Topologia do modelo Danet-FCN3. Na direita está a descrição de cada bloco [17]. (2) O bloco Ru é a unidade residual e o bloco Rut é a unidade residual transposta [18]. 47
Figura 30 – Exemplo da divisão da imagem pelo filtro de CLAHE. 50
Figura 31 – Em (A) é mostrado a região RI com todos os seus vizinhos, em (B) mostra o Pixel p do quadrante 1 da região (i, j) e sua relação aos centros das quatro regiões mais próximas. 52
Figura 32 – Em (A) é mostrado a região RB com todos os seus vizinhos, em (B) mostra o Pixel p do quadrante 2 da região (i, j) e sua relação com o centro da região mais próxima. 53
Figura 33 – Região RC com todos os seus vizinhos. 54
Figura 34 – Em (A) mostra a imagem original e em (B) mostra a imagem utilizando o filtro *CLAHE* e redimensionamento. 55
Figura 35 – Porcentagem de pixels de diferentes classes no conjunto de treinamento no banco de dados da Holanda F3. 56
Figura 36 – Histórico das épocas de treinamento da arquitetura U-Net no banco de dados Holanda F3. 57
Figura 37 – Histórico das épocas de treinamento da arquitetura Danet-FCN3 no banco de dados da Holanda F3. 58
Figura 38 – Variação do *Tile* Holanda F3. (A) - Arquitetura Danet-FCN3, (B) - Arquitetura U-Net, (C) - Arquitetura U-NET+CLAHE. 59
Figura 39 – Máscara criada a partir da arquitetura U-Net + CLAHE. 59
Figura 40 – Máscara criada a partir da arquitetura Danet-FCN3. 59
Figura 41 – Variação do tamanho dos *Tiles* no banco de dados de Penobscot. (A) - Arquitetura U-Net, (B) - Arquitetura U-Net + CLAHE. 61
Figura 42 – Histórico das épocas de treinamento da arquitetura U-Net+CLAHE no banco de dados de Penebscot 62
Figura 43 – Resultado das máscaras criadas pelas arquiteturas (A) - U-Net + CLAHE e (B) U-Net. 62

Lista de tabelas

Tabela 1 – Comparação entre trabalhos correlatos ao tema de interesse. 14
Tabela 2 – Valores de mIoU de acordo com o tamanho dos *tiles*. 58
Tabela 3 – Informações importantes das arquiteturas utilizadas no banco da Holanda. 60
Tabela 4 – Variação do mIoU de acordo com o tamanho dos *tiles* no banco de dados de Penobscot. 60
Tabela 5 – Informações importantes das arquiteturas utilizadas no banco de Penobscot. 61

Lista de abreviaturas e siglas

CLAHE	Constrast Limited Adaptive Histogram
CNN	Convolutional Neural Network
IoU	Intersection over Union
mIoU	Mean Intersection over Union
AI	Artificial Inteligence
MLP	Multilayer Perceptron
SVM	Support Vector Machine
SLIC	Simple Linear Interative Clustering
Danet-FCN	Dual Attention Fully Convolutional Network
CNN	Convolutional Neural Network
FCN	Fully Convolutional Network
NMO	Normal Move-out
CMP	Common Mid-Point
CIP	Common Image Point
ANN	Artificial Neural Network
ReLU	Rectified Linear Unit
FC	Fully Connected
SGD	Stochastic Gradient Descent
BN	Batch Normalization
CDF	Cumulative Distribution Function

Sumário

1 INTRODUÇÃO 11
1.1 Considerações Iniciais e Motivação para o Trabalho 11
1.2 Principais Contribuições 14
1.3 Informações das Simulações 15
1.4 Organização da Dissertação 16
1.5 Trabalhos Publicados 16

2 ANÁLISE SÍSMICA 17
2.1 Processamento de Dados Sísmicos. 21
2.2 Interpretação de Dados Sísmicos 23
2.2.1 Geocorpos 24
2.2.2 Horizontes 24
2.2.3 Descontinuidades 25
2.2.4 Fácies 25
2.3 Banco de Dados 26
2.3.1 Banco de dados Holanda - F3 26
2.3.2 Banco de dados Penobscot 28

3 REDES NEURAIS 30
3.1 Redes Neurais Convolucionais 31
3.1.1 Treinamento de Redes Neurais 36
3.1.1.1 Algoritmo de Retropropagação 39
3.1.2 Regularização 40
3.1.3 Aprendizagem Residual 42
3.2 Segmentação Semântica 43
3.3 U-Net 45
3.4 DANET-FCN3 46

4 PREPARAÇÃO DO BANCO DE DADOS 48
4.1 Equalização de Histograma 48
4.2 Equalização de Histograma Adaptativo Limitada por Contraste (CLAHE) 50
4.2.1 Cálculo da Função de Mapeamento 51
4.2.2 Combinação de Funções de Mapeamento 52
4.3 Pré-processamento dos dados 54

5 RESULTADOS E DISCUSSÕES 57
5.1 Banco de dados - Holanda F3 57
5.2 Banco de dados - Penobscot 60

6 CONCLUSÃO 63
6.1 Considerações Finais 63
6.2 Trabalhos Futuros 64

REFERÊNCIAS 66

1 Introdução

1.1 Considerações Iniciais e Motivação para o Trabalho

A utilização de aprendizado de máquina e técnicas de Inteligência Artificial (IA) vem crescendo exponencialmente em diversas áreas [19]. O avanço das aplicações utilizando visão computacional, principalmente em relação a aprendizagem profunda, vem conseguindo cada vez mais cobrir diversas lacunas que existiam entre as capacidades humanas e de máquinas [20]. Com essa incrível evolução, dia após dia, as máquinas se aproximam de conseguir realizar atividades de forma similar aos seres humanos, ou seja, perceber e entender pequenos detalhes, podendo utilizar o seu conhecimento adquirido para diversas aplicações, como por exemplo: reconhecimento de imagens e vídeos, análise e classificação de imagens, recriação de mídias, processamento de linguagem natural, etc. [21]. O crescimento da popularidade dos algoritmos de aprendizagem de máquina é devido a três fatores: computadores mais potentes, novas técnicas de treinamento utilizando redes neurais profundas, e muitos bancos de dados disponíveis [22].

A interpretação de dados sísmicos *offshore* tem um papel fundamental na exploração de poços petrolíferos, sendo indispensável para os geocientistas e para as descobertas de hidrocarbonetos acumulados em reservatórios. Entretanto, a interpretação de dados sísmicos é uma atividade humanamente custosa, pois exige muito tempo para análise e processamento desses dados [21]. A caracterização dos reservatórios petrolíferos e de gás se tornaram mais complexas com a chegada dos dados com informações de volumes (3D) e atributos sísmicos. Pois, com esses dados são fornecidas, também, informações qualitativas da geometria e dos princípios físicos das camadas litológicas [23]. Os dados sísmicos são obtidos por meio de ondas sísmicas captadas por hidrofones, adquiridas de maneira indireta e com isso possuem uma grande quantidade de ruído, dificultando a interpretação. A reconstrução de um dado sísmico é um enorme desafio, mesmo com o avanço tecnológico na aquisição e processamento de dados. Os prazos cada vez menores para as empresas serem mais competitivas e a quantidade de dados cada vez maior também são fatores complicantes [24].

Sendo assim, o uso de ML (*Machine Learning*) vem se tornando cada vez mais frequente, visando aumentar a qualidade dos resultados e em prazos pequenos. O uso de ML em aplicações comerciais referentes a segmentação de dados sísmicos, que é a criação de uma máscara com as camadas litológicas separadas por classes, possuem abordagens que podem ser não supervisionadas ou supervisionadas [19]. Os modelos de aprendizado de máquina não supervisionados são uma classe de algoritmos que podem descobrir padrões e relacionamentos em potencial nos dados sem qualquer referência ou rotulação (*label*).

Aplicando esses algorítmicos pode-se identificar novas relações entre atributos sísmicos anteriormente desconhecidos, ajudando assim os especialistas a gerar explicações mais precisas. A grande quantidade de dados sísmicos e a falta de certeza entre as camadas litológicas contribuem como mais uma vantagem para a aplicação desses modelos [25]. Na utilização da abordagem não supervisionada são comumente utilizados algoritmos de agrupamento.

Entretanto, essas abordagens não são capazes de resolver todos os problemas de classificação ou segmentação em imagens sísmicas. Em Barnes e Laughlin [26] foi mostrado que a acurácia da abordagem depende diretamente dos atributos selecionados e não tanto do método de agrupamento utilizado. A escolha do conjunto entre as centenas ou milhares de atributos disponíveis acaba se tornando um problema de dimensionalidade dos dados, pois nem todo atributo é relevante para uma análise mais profunda e a maioria possui informações redundantes, sendo muito imporante a seleção de variáveis [27].

Com o surgimento da aprendizagem profunda (do inglês, *Deep Learning*), estudos foram realizados para aplicação dessa técnica como alternativa para solucionar as dificuldades encontradas pelas técnicas de classificação manual e assistida no aprendizado não supervisionado nas fácies sísmicas [28]. Esse tipo de algoritmo é utilizado para extrair uma representação dos dados para realizar a classificação das fácies sísmicas de forma automática. Para tarefas de classificação de dados sísmicos, muitos modelos de aprendizado de máquina supervisionados têm sido explorados, como máquinas de vetores de suporte (SVM) [18], árvores de decisão [29], redes neurais perceptron multicamadas (MLP) ou redes neurais convolucionais [18], [29]. Esses tipos de algoritmos usam rótulos predefinidos nos dados sísmicos, por humanos que conhecem a área. Portanto, seu sucesso depende dos rótulos dos dados sísmicos e do conhecimento geral do especialista.

Os algoritmos de aprendizado de máquina aplicados a dados sísmicos podem, ainda, ser divididos em duas categorias: algoritmos clássicos de agrupamento e algoritmos de segmentação de imagem. Algoritmos clássicos de agrupamento têm sido aplicados com sucesso no campo da exploração de dados sísmicos. Por exemplo, Mapas de Kohonen (*Self Organizing Maps*) [24], *k-Means* [25], *Fuzzy c-Means* [30] entre outros.

Nos algoritmos de segmentação de imagens, as camadas sísmicas são tratadas como sequências de imagens processadas independentemente. Portanto, uma etapa de pós-processamento é necessária para combinar os resultados em volumes sísmicos. Outro trabalho utilizando aprendizagem não supervisionada através das técnicas de segmentação utilizando *Watershed* e *Simple Linear Iterative Clustering - (SLIC)* é apresentado em [27]. Essas redes foram usadas para criar uma máscara para dados sísmicos sem rótulos pré-definidos. Neste artigo [27] nenhuma técnica de pós-processamento foi necessária para atingir um resultado satisfatório. Outro trabalho importante na área foi a investigação de diversos filtros em dados sísmicos [31].

Civitarese em [32] conseguiu realizar com sucesso a criação de máscaras de saída a partir de dados sísmicos de entrada utilizando uma rede Danet-FCN (*Dual Attention Fully Convolutional Network*), que é uma arquitetura baseada em redes convolucionais. Foram utilizados 2 bancos de dados sísmicos, o primeiro foi o de Penobscot, da Bacia da Escócia, e o outro da Holanda da Bacia de Central-Graben. Durante o treinamento da rede, ela foi capaz de alcançar uma acurácia de pixel da ordem de 97 %. Nesse artigo os autores não utilizaram nenhuma técnica de segmentação semântica de imagem, porém, chegaram em um *F1-Score* de até 98,7% , utilizando menos de 5% do banco de dados para treinamento. Entretanto, das 8 classes, as classes 3 a 7 foram agrupadas em apenas uma classe, tornando-se assim um problema de classificação com apenas 4 classes [33].

Em relação à segmentação semântica, Shale-Shwartz em [34] utilizou uma arquitetura de rede neural profunda especialmente projetada para segmentar semanticamente imagens sísmicas com uma quantidade mínima de dados de treinamento. Nesse trabalho são comparados dois métodos de rede neural, a FCN e U-Net [17, 35]. Foi mostrado que essa abordagem pode atingir mais de 99% de intersecção sobre a união (IoU). Além disso, os resultados qualitativos mostram que o modelo obtido pode produzir máscaras muito próximas da interpretação humana. Um estudo foi publicado sobre a classificação das fácies sísmicas e como se pode dividi-las para a melhor compreensão por uma rede neural [33]. Também foi mostrado que é possível utilizar transferência de aprendizado (*Transfer Learning*) para acelerar e melhorar o treinamento das redes [36]. E como trabalho posterior, conseguiram aplicar essas técnicas para a realização da segmentação semântica das camadas rochosas [37].

Entretanto, em todos os artigos citados acima faltam informações sobre os ajustes dos hiper-parâmetros, tais como, o otimizador utilizado, a taxa de aprendizagem etc. Além do fato de que a rede que possui melhor desempenho em classificação de imagens sísmicas (Danet-FCN3), possui uma arquitetura complexa, utilizando diversas camadas de convoluções e convoluções transpostas, resultando em alta complexidade computacional. Outras arquiteturas, tais como a U-Net, são bem menos complexas, porém, possuem desempenho menor na classificação das imagens sísmicas [37].

Diante da literatura sobre segmentação semântica de dados sísmicos, foram levantadas algumas informações importantes dos artigos, como mostra a Tabela 1. Com isso, foi decidido realizar um estudo para tentar melhorar o mIoU utilizando a U-net com um pré-processamento.

O uso de redes neurais para segmentação de imagens é uma tarefa que precisa de estudos mais aprofundados para encontrar um ponto de equilíbrio entre nível de complexidade computacional e desempenho, que não é uma tarefa trivial. Com isso, viu-se necessária a melhoria da arquitetura de redes neurais de baixa complexidade, tais como a U-Net, a fim de encontrar um cenário ótimo para realização desta tarefa.

Tabela 1 – Comparação entre trabalhos correlatos ao tema de interesse.

Autor	Número de Imagens	Métrica Utilizada	Quantidade de Classes	Tamanho do *tile*	Arquitetura
Daniel Civitarese [38]	412	98,25 % (mIoU)	7	40x40	Danet-FCN
Daniel Civitarese [38]	276	97,00 % (Precisão)	6	40x40	Danet-FCN3
Daniel Civitarese [33]	160	75,80 % (F1-Score)	4	40x40	CNN
Bilal Abid [39]	13421	93,31 % (mIoU)	7	128x128	DeepLab v3Plus
Zengyan Wang [40]	41	88,30 % (mIoU)	Não Mencionado	Não Utilizado	U-Net + Attention
Reinaldo Silva [13]	Não Mencionado	98,00 % (mIoU)	6	64x25	CNN
Daniel Civitarese [32]	100	99,00 % (mIoU)	7	80x120	Danet-FCN3
Maykol Trinidad [6]	1102	89,50 % (FWIU)	6	Não Utilizado	Atrous U-Net
Xiaoyu Chen [41]	1102	77,50 % (mIoU)	6	Não Utilizado	HRNetV2 -W32
Danilo Calhes [42]	32	89,16 % (mIoU)	7	64x64	PSP-Net
Trabalho Proposto	100	90,02 % (mIoU)	6	256x256	U-Net + CLAHE

No trabalho [42] conseguiu-se chegar em um resultado bom, com o desempenho de quase 90% de mIoU na arquitetura proposta, porém, é utilizado uma arquitetura complexa, a PSP-Net. Os trabalhos [6], [41] e [39] obtiveram resultados consideráveis com as arquiteturas proposta, porém, foram utilizado uma grande quantidade de imagens para os treinamentos

1.2 Principais Contribuições

O processo de interpretação de dados sísmicos é limitado por alguns aspectos, tais como o tempo necessário para as análises dos especialistas, incertezas devido à subjetividade da análise e as limitações do software utilizado para interpretação. O processo deve integrar totalmente os dados geológicos, geofísicos, petrofísicos e de engenharia. Lidar com grandes quantidades de dados pode ser um desafio, mesmo para geólogos experientes. Além disso, caso o especialista proponha uma solução abaixo do ideal, isso pode causar um prejuízo de milhões de dólares em perdas durante a exploração de petróleo e gás [25].

Com isso, este trabalho apresenta um estudo cuidadoso para a projeção de dois modelos de rede neurais para a realização da tarefa de segmentação semântica em imagens

sísmicas. Dessa forma, são esperadas as seguintes contribuições:

- (1) - Criação de redes neurais artificias treinadas e ajustadas para a realização da segmentação semântica no banco de dados sísmicos proposto. A alta qualidade da rede treinada pode ser comprovada através da métrica de *Intersection Over Union* (IoU) no grupo de teste realizado no banco de dados.
- (2) - Demostração da importância do uso de técnicas de pré-processamento para a melhoria da arquitetura da rede.
- (3) - Verificar a importância do tamanho do *title* no desempenho das arquiteturas estudadas.

Além do estudo em relação às arquiteturas utilizadas para a segmentação semântica de imagens sísmicas apresentado, este trabalho traz a contribuição de ser um experimento facilmente replicável, o que poderá servir como base para novos trabalhos, tendo em vista que os bancos de dados utilizados são públicos.

O objetivo desta dissertação é implementar um segmentador de fácies sísmicas empregando redes convolucionais, realizar adições na etapa de pré-processamento para verificar a melhoria da rede e compará-la com as redes consideradas como estado-da-arte encontradas na literatura.

Além disso, na metodologia aplicada, é possível realizar experimentações dessa arquitetura ou de arquiteturas semelhantes em outros *datasets*, tendo em vista uma redução de tempo de especialista analisando os dados e aumento da acurácia na etapa de caracterização dos reservatórios de petróleo e gás.

1.3 Informações das Simulações

Foram realizadas simulações utilizando a linguagem de programação *Python* para desenvolver as arquiteturas de *Machine Learning* e realizar o treinamento dos modelos. Para facilitar a implementação das topologias das redes, foram utilizados as seguintes bibliotecas para o uso de inteligência artificial e manipulação de dados: Pandas v1.0.1, TensorFlow v2.12.0, Keras, OpenCV v4.7.0, NumPy v1.24.3, Matplotlib v3.7.1, Glob v2.1.1, OS v17.2.0, e Tifffile v2023.4.12.

Além, disso, todos os testes foram realizados em um computador com as seguinte especificações: Processador I7 6800K, memória ram 16Gb ddr4, placa de video GTX1070 Asus strix.

Para a realização das interpretações dos resultados, foram utilizadas algumas métricas de avaliação, como:

- IoU - A média da razão entre a área da intercessão e da área da união de uma imagem sísmica, como mostra a Figura 1

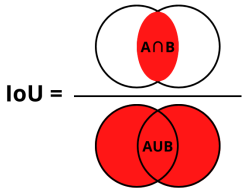


Figura 1 – Descrição do cálculo do mIoU

- mIoU - É a Média dos IoU de um conjunto de imagens.
- Precisão - É a porcentagem de classes acertadas pela arquitetura.

Com essas métricas, foi possível realizar a comparações entre os resultados e avaliar se houve a melhoria na arquitetura proposta nessa dissertação.

1.4 Organização da Dissertação

No Capítulo 2 está a descrição do que é um dado sísmico, suas propriedades, como é adquirido e preparado, e também quais bancos de dados foram estudados. No Capítulo 3 é explicado os modelos de redes neurais para aprendizado profundo, como as arquiteturas que foram utilizadas nas simulações, a U-Net e Danet-FCN3. No Capítulo 4 é explicado como foi feita a preparação do banco de dados para as simulações e como foi realizado o pré-processamento nas imagens. No Capítulo 5 são apresentados os resultados das simulações realizadas e é feita uma comparação como o estado-da-arte. Por fim, no Capítulo 6 é apresentado a conclusão do trabalho com todos os pontos relevantes levantados durante a pesquisa.

1.5 Trabalhos Publicados

Durante o desenvolvimento deste trabalho, foram publicados dois artigos, um em um congresso nacional e outro em uma conferencia internacional, são eles:

- "Análise de Arquiteturas de Redes Neurais Para Segmentação Semântica de Dados Sísmicos- XL Simpósio Brasileiro de Telecomunicações e Processamento de Sinais (SBrT 2022).
- "*Analysis of Neural Network Architectures for Semantic Segmentation of Seismic Data*- trabalho aceito para publicação na *Computing Conference 2023*.

2 Análise Sísmica

A sísmica é a base da Sismologia (Geofísica) que visa a compreensão do interior da terra através da análise do movimento do solo [34]. Inicialmente os sismólogos descobriram que as ondas sísmicas geradas pelos terremotos continham informações importantes sobre o interior da terra (crosta, manto e núcleo). Essas mesmas ondas poderiam ser geradas artificialmente, em um nível menor de intensidade que as provenientes de terremotos, com o intuito de se interpretar a estrutura superficial da terra, localizar minerais, recursos hídricos e petrolíferos. A sismologia se concentra no estudo da geração, propagação e gravação das ondas elásticas na terra e dos recursos que as produzem. Na exploração petrolífera é largamente utilizada para se identificar esses recursos e otimizar a extração de óleo e gás em reservatórios complexos [1].

A geração e propagação de ondas sísmicas é feita com a aplicação de uma fonte externa de força sobre uma superfície. Essa força dá origem a deformações que produzem ondas elásticas que se propagam para as subcamadas da região onde a força foi aplicada. Para a geração de ondas sísmicas, é necessária a utilização de uma fonte mecânica impulsiva de energia. A fonte pode ser natural (como um terremoto) ou artificial (como canhões de ar ou dinamite) [1]. Quando criada, a onda se deforma em sua origem, resultando em outras deformações adjacentes que carregam uma parte da energia da deformação anterior, repetindo esse processo até que toda a energia inicial da onda seja dissipada no meio na forma de calor [43]. A onda é propagada em todas as direções possíveis, durante o percurso da onda sísmica para os meios mais internos de uma área, parte da onda é refletida quando atravessa a interface entre as camadas rochosas, como mostra a Figura 2. Essa reflexão é capturada por sensores posicionados na superfície. Cada sensor é responsável por registrar as oscilações da intensidade das reflexões de ondas captadas e o seu tempo de chegada. Cada sensor dá origem a um traço sísmico (conjunto e valores de impedância de onda associados a um sensor) e ao conjunto de traços se dá o nome de seção sísmica [44].

Na maioria das vezes, para a aquisição de dados sísmicos, são utilizadas centenas de receptores tanto em terra como em mar. Os receptores são organizados equidistantes entre si em uma linha reta e cada receptor dará origem a um traço sísmico. A cada linha se dá o nome de linha sísmica ou seção sísmica. Quando a aquisição de dados sísmicos é feita utilizando-se uma única linha se diz que é bidimensional (2D), pois o dado resultante é composto pelo tempo que a onda sísmica levou para percorrer o trajeto até a interface entre as rochas e o valor da magnitude da reflexão da onda. Em contrapartida, nos casos onde muitas linhas sísmicas são utilizadas, a aquisição é dita tridimensional (3D) [45].

A Figura 3 exibe o *layout* de aquisição em terra. Na parte (a), há um diagrama

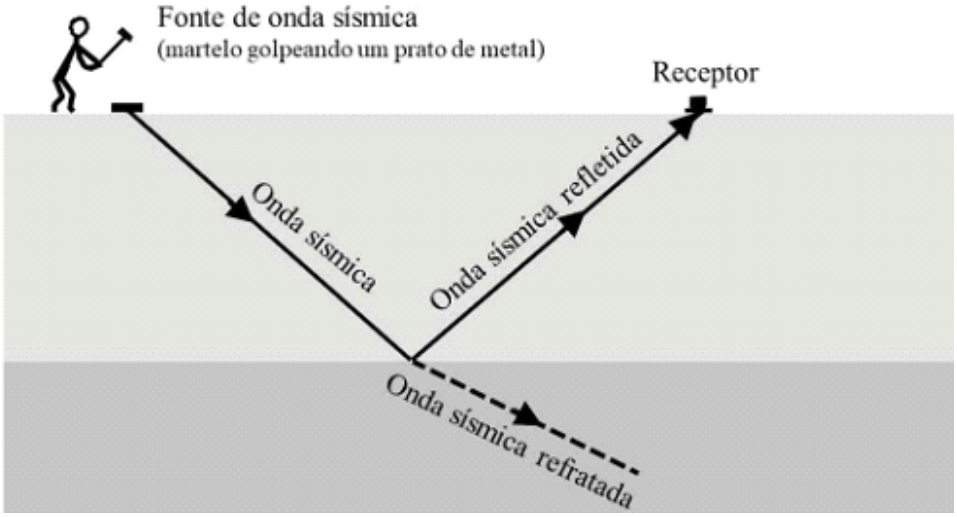


Figura 2 – Geração de ondas sísmicas [1].

mostrando o *layout* da energia transmitida e refletida do ponto de disparo e os receptores. Em todas as interfaces, uma parte da energia é transmitida e outra é refletida (estas alcançam os receptores na superfície). Existem duas interfaces refletoras, numeradas (1) e (2) [7].

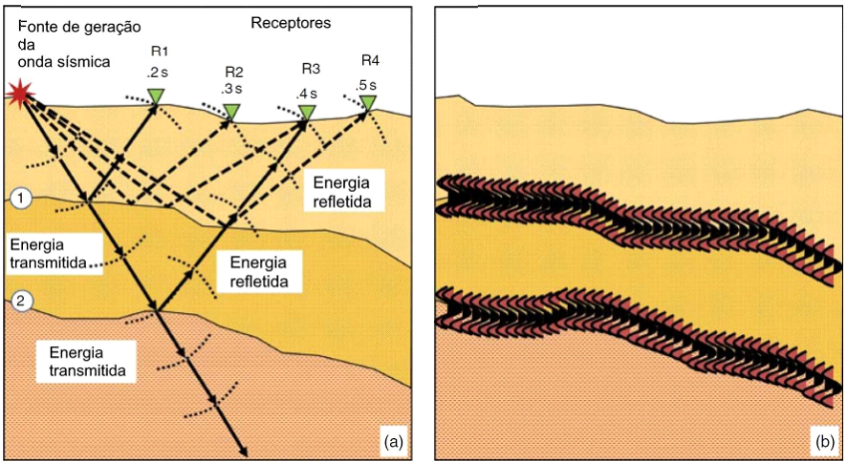


Figura 3 – Leitura de aquisição em terra. Rn: número do receptor; em (b) traços sísmicos simplificados correspondentes às interfaces em (a) [2].

A aquisição de dados sísmicos terrestres, como mostra a Figura 4, pode ser realizada em qualquer parte do solo terrestre exposto na superfície. Porém, esse tipo de aquisição pode sofrer de problemas de logística como área de difícil acesso ou proibidas, altas atividades que produzem ondas elástica no solo gerando muito ruído durante a captura dos sensores etc. As fontes mecânicas de energia comumente utilizadas nessas áreas incluem explosivos (como dinamite ou cordas explosivas), vibradores, canhões de ar e queda de peso. A aquisição marinha de dados sísmicos, podem ser adquiridos em locais rasos (1-40 m), ou em até águas ultraprofundas (>1500 m). Para altas laminas d'água são utilizados

navios especializados com matrizes de hidrofones (receptores) rebocadas por um ou mais cabos que são chamados *streamers*, como se pode ver na Figura 5 [45].

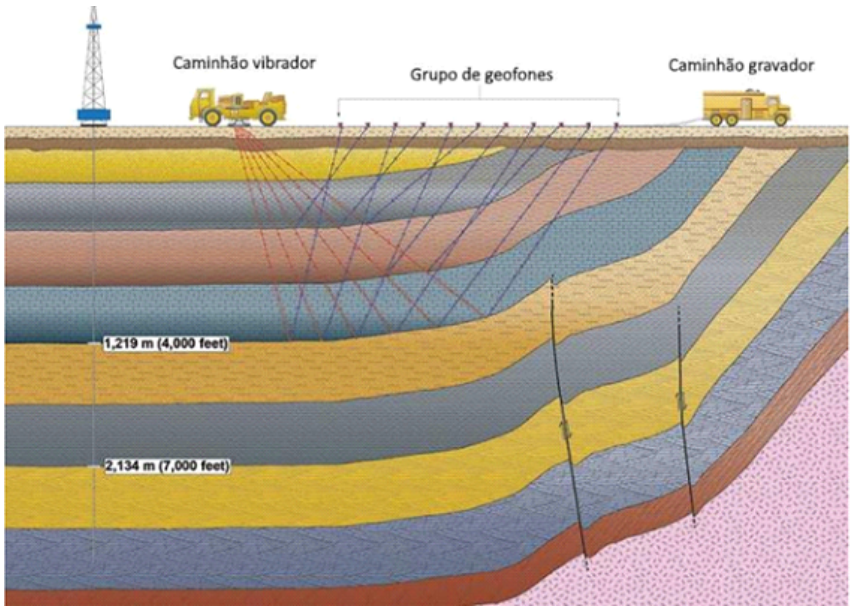


Figura 4 – Esquema de aquisição sísmica em terra [3].

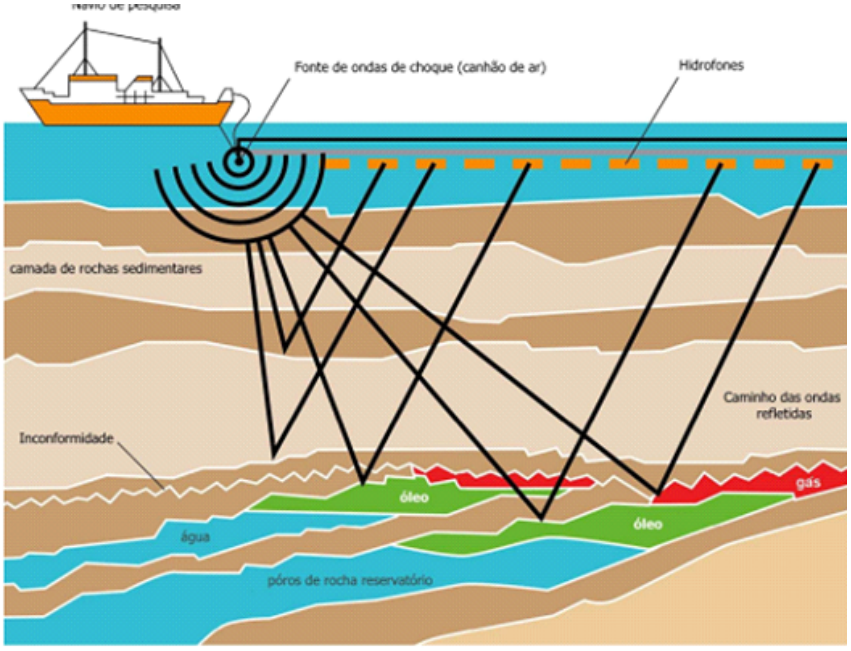


Figura 5 – Aquisição sísmica no mar [4].

Após a aquisição sísmica, é feito o processamento de dados sísmicos, o qual é uma sequência de processos matemáticos e estatísticos executados nos dados brutos da aquisição, a fim de se extrair informações voltadas a interpretações sísmicas nas quais o geofísico possa identificar estruturas geológicas favoráveis a acumulação de hidrocarbonetos [44]. O objetivo principal é corrigir os sinais sísmicos através da diminuição de ruído (como sinais

captados do ambiente), correção do tempo de viagem de reflexão e correção da forma e posição das ondas refletidas.

Os resultados do processamento são dados sísmicos chamados de *post-stack*, onde foram feitas melhorias no sinal de ondas sísmicas e correções nas posições dos eventos de reflexão pela sísmica com o intuito de se identificar regiões com algum significado geológico.

A interpretação busca extrair todas as informações de subsuperfície, disponíveis dos dados sísmicos processados, assim como a informação estrutural, estratigráfica, propriedades de rochas subterrâneas, velocidade de propagação de onda, tensão etc. Esse processo requer uma aquisição sísmica com alta resolução, conhecimento da geologia local e experiência do intérprete. Durante a interpretação se verifica a existência de padrões de reflexão para que os mesmos sejam associados a alguma propriedade sísmica. A interpretação é feita através da seleção dos topos (início) dos padrões e da correlação entre esses topos os quais podem formar uma área ou volume [46].

A interpretação estrutural é comumente realizada durante a fase inicial da exploração de reservatórios de petróleo, onde se verifica se há uma probabilidade alta da existência de hidrocarbonetos que justifique os investimentos monetários a serem realizados. A Figura 6 contém um exemplo de interpretação sísmica.

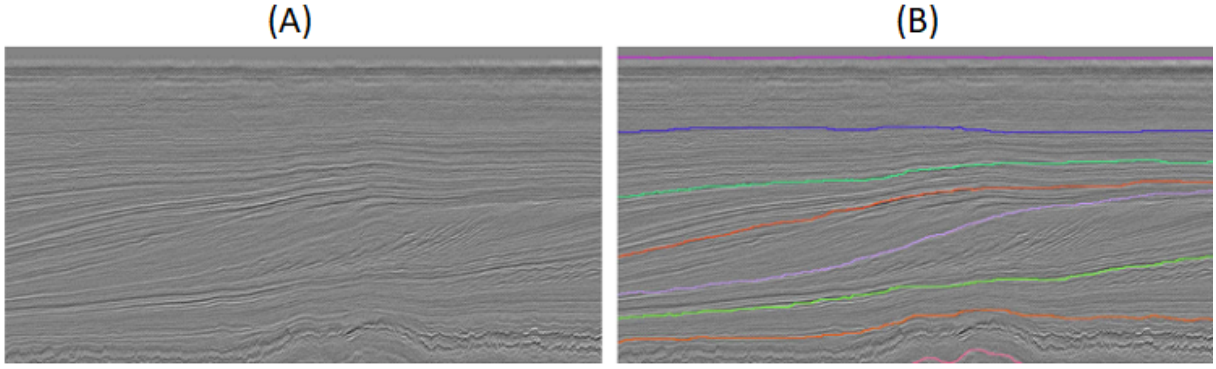


Figura 6 – A imagem (A) mostra um dado sísmico cru. As linhas coloridas em (B) representam as geometrias identificadas na seção sísmica [5].

A análise estrutural é o estudo da geometria das interfaces das rochas a partir dos tempos de reflexões. Segundo [47], a sua principal aplicação é a busca de armadilhas estruturais (trapas) que possam conter hidrocarbonetos. Os eventos de reflexão de interesse são marcados utilizando-se cores e rotulados com algum nome, até que seu significado geológico seja estabelecido através da correlação com outros dados, tais como, poços que contém informações de litologia associados a profundidade.

Os dados sísmicos processados podem ser representados em três tipos de seções sísmicas: *inline*, *crossline* e *depth-slice* ou *time*. A seção *inline* é formada a partir de valores fixos nos eixos Y e Z de um volume sísmico, e geralmente é paralelo à direção da linha de aquisição de dados sísmicos. A seção *crosline* é formada a partir de valores fixos nos eixos

X e Z de um volume sísmico, geralmente é perpendicular à direção de linha de aquisição de dados sísmicos, como mostra a Figura.7.

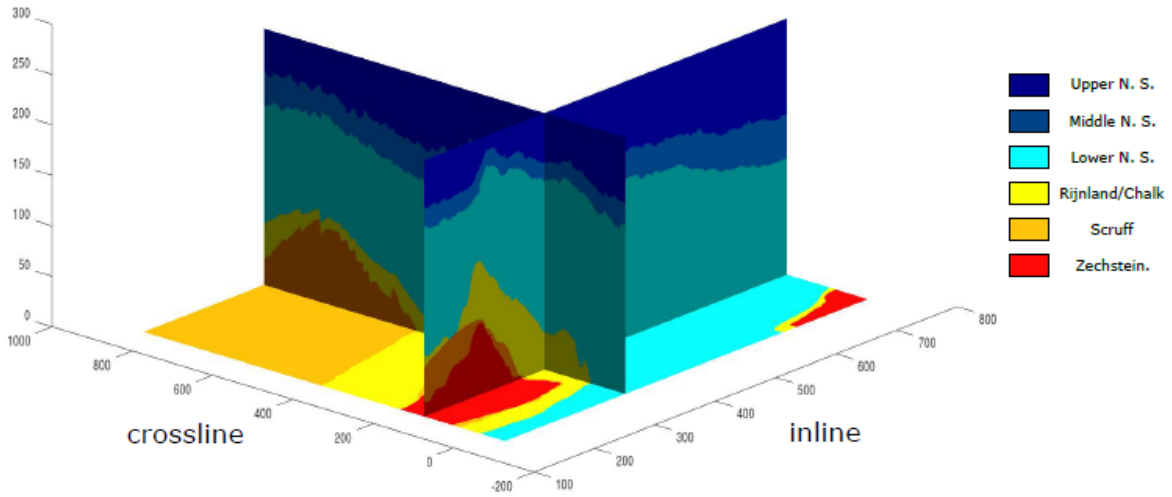


Figura 7 – Diferença entre *Inline* e *Crossline* no banco de dados da Holanda F3 [6].

Já a seção *depth-slice* é formada a partir de um corte transversal das *inlines* e *crosslines*. Uma analogia para se entender uma seção sísmica é vê-la como um plano que compõe o volume sísmico, como exibido na Figura 8, onde várias seções se entrecortam [7].

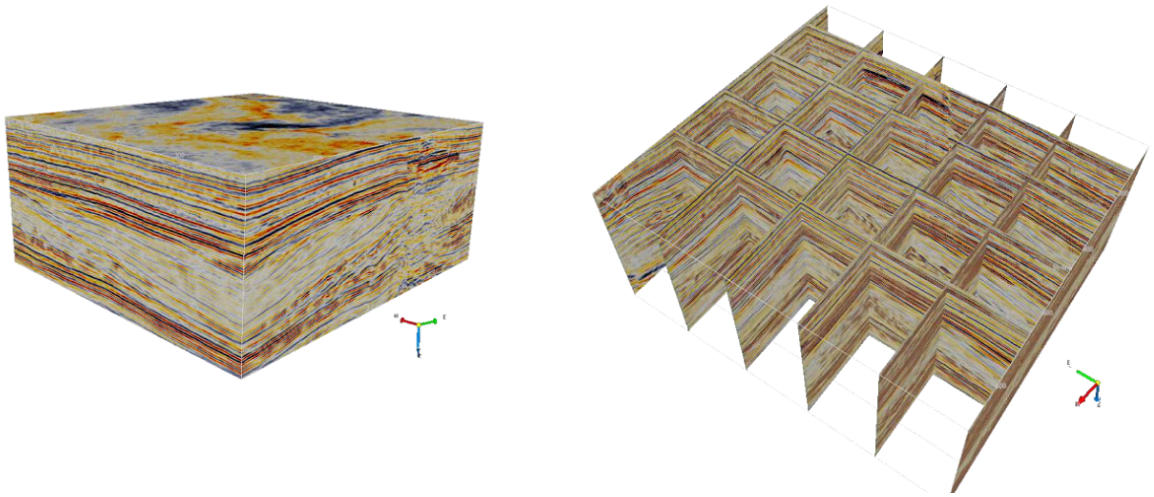


Figura 8 – Exemplo de algumas seções sísmicas que compõem o volume sísmico do banco de dados da Holanda F3 [7].

2.1 Processamento de Dados Sísmicos.

Os geofísicos chamam os dados brutos coletados pelos receptores de "formato pré-stack". Estes dados são agrupados como "gather", como mostra a Figura 9, que é

uma classe de traços sísmicos com características geométricas similares. E então, a análise de dados pré-stack geralmente envolve o uso de pontos de imagem comum (*Common Image Point - CIP*) ou ponto médio comum (*Common mid-point - CMP*) para examinar atributos importantes, como amplitude, conteúdo de frequência, e ruído de fase.

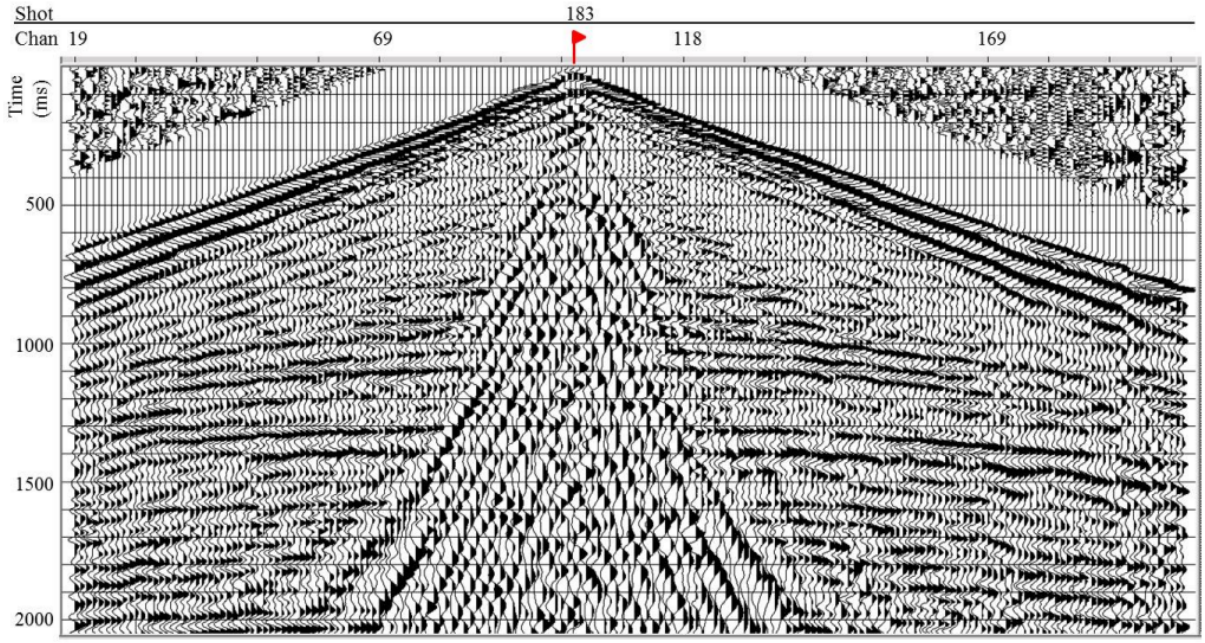


Figura 9 – Exemplo de uma imagem sísmica "gather" [8].

Existem várias técnicas disponíveis para processar "gathers"sísmicos e convertê-los em imagens tridimensionais precisas da subsuperfície. Duas das técnicas mais amplamente utilizadas são a migração e a correção do sobre-tempo normal (*Normal Move-Out - NMO*) com *stacking*.

A migração envolve a manipulação da hipérbole de reflexão das ondas sísmicas como uma hipérbole de difração. Isso permite que o agrupamento de pontos de imagem comum (CMP) seja migrado para sua posição correta na subsuperfície. É necessário experimentar com diferentes velocidades para avaliar os resultados e determinar a velocidade média. Quando a velocidade de migração é igual à velocidade média, a hipérbole de difração colapsa em seu pico, o que aparece no traço de deslocamento zero.

A correção do sobre-tempo normal (NMO) é um processo que corrige o tempo de viagem medido de uma onda sonora refletida para o tempo de viagem na incidência normal. Isso é feito usando uma equação que relaciona o tempo de viagem medido, o deslocamento e a velocidade da onda. A correção NMO é importante porque permite que os traços sísmicos sejam comparados corretamente, independentemente da posição da fonte e do receptor.

Em conjunto, essas técnicas são essenciais para a construção de imagens precisas e detalhadas da subsuperfície, que são amplamente utilizadas na indústria petrolífera, na

prospecção mineral e em aplicações geotérmicas.

A correção sobre-tempo normal (NMO) é um processo através do qual podemos corrigir o tempo de viagem medido de uma onda sonora refletida t em um determinado deslocamento x, para obter o tempo de viagem na incidência normal t_0 usando a equação:

$$t_0^2 = t^2 - \frac{x^2}{u_{NMO}^2} \tag{2.1}$$

onde u_{NMO} é a velocidade do NMO. Existem muitas variações desta equação, mas esta é a sua forma mais simples e quando aplicada a uma seção CMP, transforma a hipérbole criada pela reflexão em uma linha reta horizontal. É baseado na suposição de que em uma reunião CMP, os tempos de viagem de reflexão como uma função de deslocamento seguem trajetórias hiperbólicas e é capaz de remover o efeito de deslocamento nos tempos de viagem. Em seguida, os traços em cada coleta CMP são ajustados para formar um traço empilhado em cada localização de ponto médio. Como resultado dessa correção de movimento, os traços são estendidos de uma maneira variável no tempo, fazendo com que sua frequência se desloque para a extremidade inferior do espectro, como mostra a Figura 10 [25].

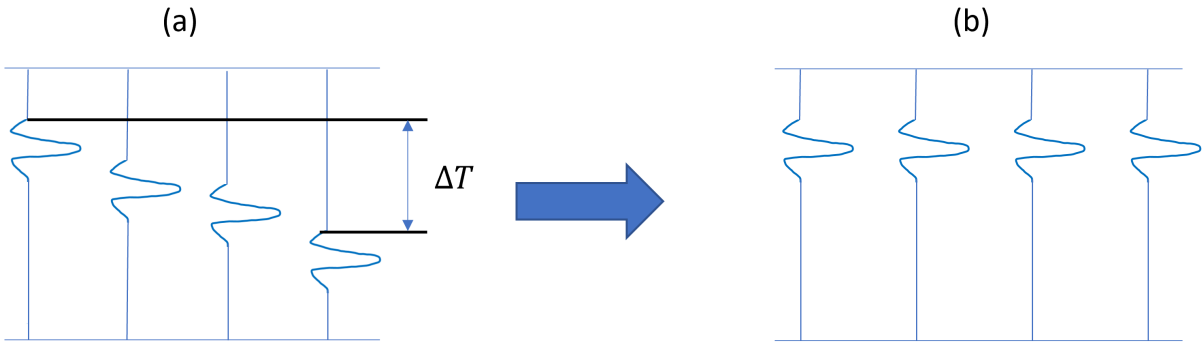


Figura 10 – NMO e *stacking* [8].

A Figura 11 ilustra as duas técnicas lado a lado. A imagem (a) mostra uma coleta de CMP de um único refletor em um meio de velocidade constante. A imagem (b) mostra o espectro de velocidade obtido pela migração usando várias velocidades constantes e exibindo o deslocamento zero para cada uma e, finalmente, a imagem (c), o espectro de velocidade derivado pela correção NMO e *stacking* [25].

2.2 Interpretação de Dados Sísmicos

Após a conclusão da etapa de processamento, tem-se as imagens sísmicas dos dados. O próximo passo é a interpretação desses dados, que requer que um intérprete construa o modelo geológico mais preciso possível. Isso é realizado por desenho manual ou usando

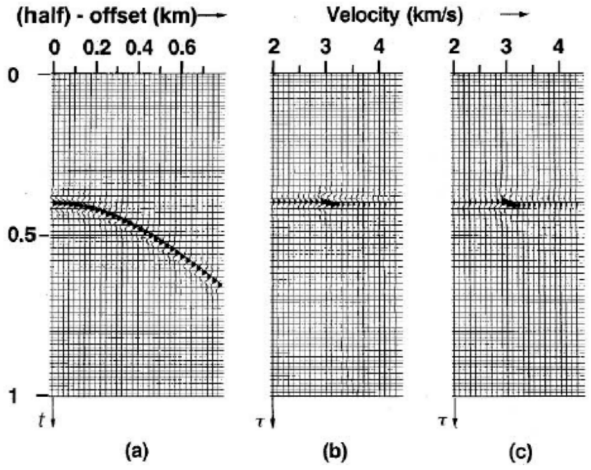


Figura 11 – técnicas de processamento "gather" [9].

um cursor na tela, com base na compreensão geológica do intérprete e na seleção da interpretação mais viável dentre as várias possíveis sugeridas pelos dados.

A subsuperfície possui inúmeras características geológicas relevantes, mas neste capítulo será abordado apenas aquelas que contribuem para a interpretação visual dos dados, ao invés da análise quantitativa. Aqui estão algumas das características geológicas mais comuns usadas na interpretação de dados sísmicos.

2.2.1 Geocorpos

Os geocorpos sísmicos são características sísmicas que se baseiam em relações matemáticas envolvendo a geometria e propriedades físicas da subsuperfície, resultando em vários atributos distintos de interesse. Um exemplo destacado de geocorpos são os domos e diapiros de sal, que são estruturas volumosas de sal subterrâneo e desempenham um papel vital na busca de hidrocarbonetos. No entanto, a interpretação do sal pode ser bastante difícil, já que está associado a reflexões fracas e é difícil diferenciá-lo de outros sedimentos. Outro tipo de geocorpo é o canal sísmico, uma depressão linear com base côncava por onde água e sedimentos podem fluir e se depositar em corpos únicos. Eles podem ter diversas formas, como retas, sinuosas ou ramificadas. Tanto os corpos de sal quanto os canais são excelentes indicadores de depósito de hidrocarbonetos, pode se ver esses exemplos na Figura 12 [25].

2.2.2 Horizontes

Uma característica de horizonte geralmente se refere a uma superfície de camadas de sedimentos ou rochas vulcânicas dispostas em camadas, separadas por mudanças na superfície estratigráfica, sejam elas litoestratigráficas ou cronoestratigráficas. Do ponto de vista geológico, existe sempre incerteza quanto à localização precisa dos horizontes e a

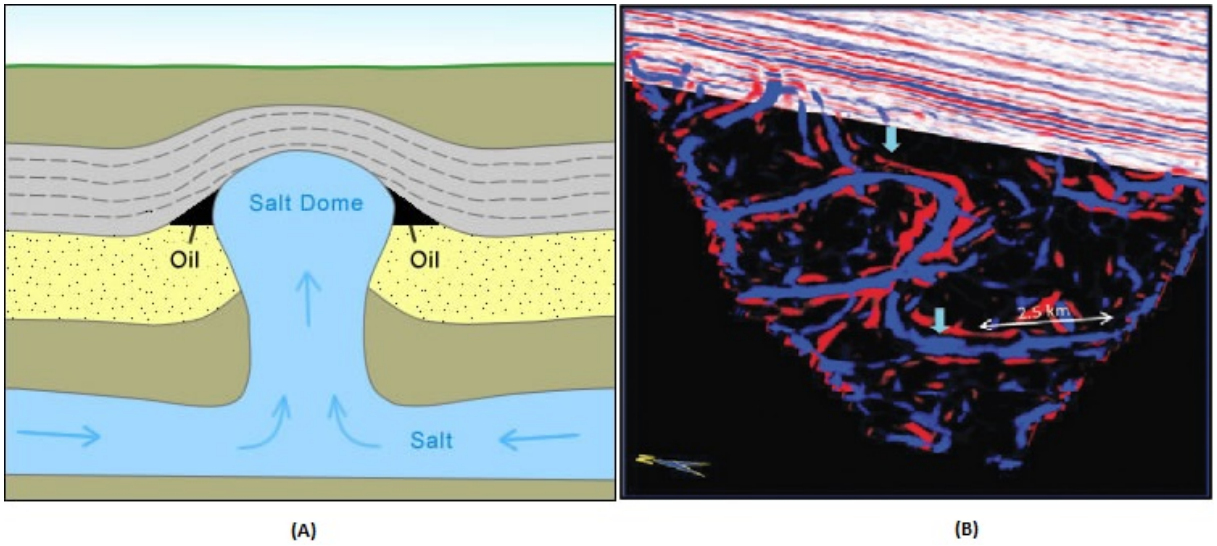


Figura 12 – (A) Domo de sal [10] e (B) Canais sísmicos (na cor azul) [11].

verdadeira relação entre eles e as superfícies geológicas correspondentes é considerada um dos problemas não resolvidos na geofísica de exploração [25].

2.2.3 Descontinuidades

As descontinuidades sísmicas são superfícies que apresentam uma variação repentina na velocidade das ondas sismográficas. Essas variações são sinais de características importantes, sendo as falhas sísmicas as mais frequentes. Uma falha é uma rachadura na rocha, onde há um deslocamento significativo devido ao movimento de massa. Normalmente, são criadas na crosta terrestre pelas forças das placas tectônicas e as maiores delas formam as zonas de subducção ou as falhas transformantes. Essas falhas podem ser causadas por terremotos ou por uma deformação sísmica ao longo do tempo. O plano da falha é a superfície que representa a rachadura de uma falha, enquanto que a trajetória da falha (ou linha de falha) é o lugar onde a falha pode ser observada e mapeada na superfície. Na Figura 13, é possível ver a representação de 5 horizontes e 2 falhas.

2.2.4 Fácies

A fácie sísmica é um termo técnico usado para descrever áreas na Terra com rochas de composições ou características similares que são agrupadas em um ambiente comum. Essas áreas são delimitadas por reflexos em imagens sísmicas e indicam divisões entre regiões que contêm sedimentos formados por diferentes tipos de processos geológicos. A fácie pode ter quaisquer características observáveis das rochas, como sua aparência geral, composição da formação e variações na área geográfica. As semelhanças entre as fácies podem ser químicas, físicas ou biológicas, que as diferenciam das rochas adjacentes. Alguns exemplos de ambientes sedimentares e fácies incluem:

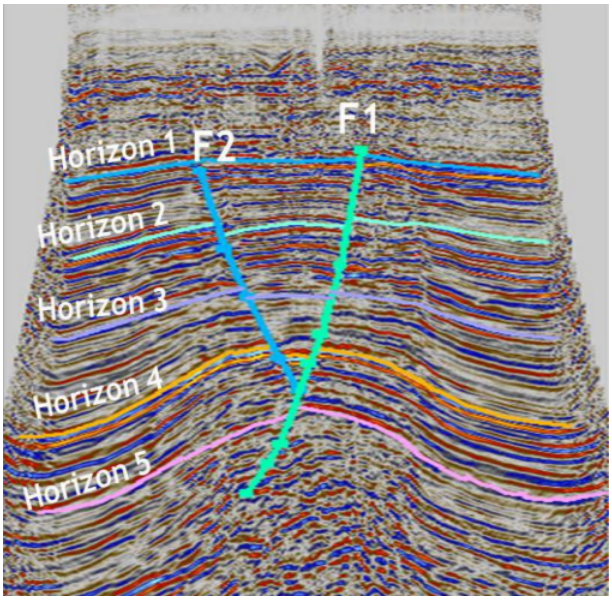


Figura 13 – Horizontes com falhas [12].

- Misturas de areia, silte ou lama colocadas em um delta em forma de leque na foz de um rio.
- Sedimentos arenosos grossos em um canal de rio sinuoso.
- Sedimentos de granulação fina em leito de lago raso.

As fácies sísmicas são as feições geológicas que serão utilizada como entradas no conjunto de dados deste trabalho e será descrita mais detalhadamente nos capítulos seguintes.

2.3 Banco de Dados

Após o tratamento realizado pelos geólogos experientes, como descrito anteriormente, é criado um modelo de bloco sísmico 3D. Esse bloco serve como base para a tomada de decisão em relação à perfuração de futuros poços. No entanto, é importante ressaltar que a confidencialidade da indústria de petróleo e gás impede o compartilhamento de conjuntos de dados. Portanto, conjuntos de dados de alta qualidade e disponíveis publicamente para interpretação sísmica são extremamente valiosos e raros. Neste trabalho serão utilizados dois bancos de dados, que são descritos a seguir.

2.3.1 Banco de dados Holanda - F3

Este conjunto de dados se originou de um artigo [13], cujo objetivo declarado é ajudar com a falta de dados sísmicos rotulados gratuitamente disponíveis, fornecendo o

conhecido conjunto de dados da Holanda F3 com rótulos para servir como referência para novos métodos de aprendizado de máquina. Em sua versão, em vez de fornecer o conjunto de dados e rótulos como cubos, eles simplesmente fornecem os *inlines* e os *crosslines*. Sua interpretação resultou na identificação de 9 horizontes, que foram nomeados como H1, H2, ..., H9, ordenados em ordem descendente de idade geológica, e as imagens rotuladas são criadas pela interseção entre as linhas sísmicas e as superfícies dos horizontes, como mostra na Figura 14.

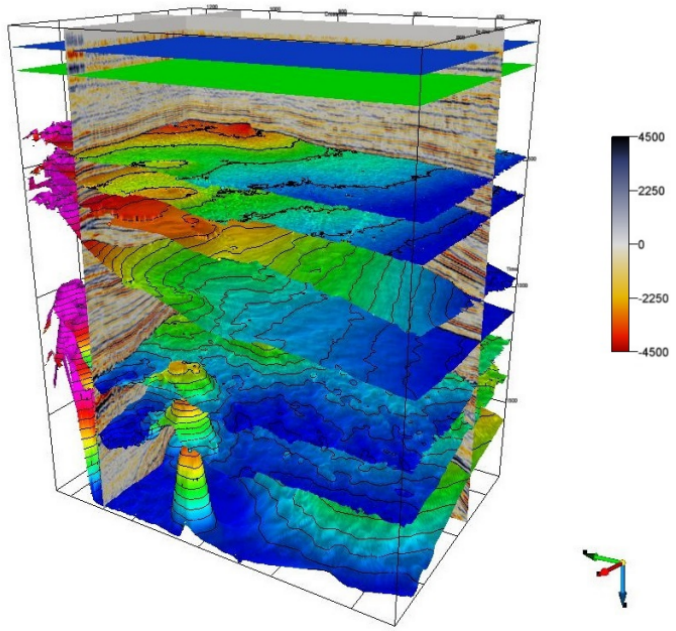


Figura 14 – Nove horizontes interpretados mostrados junto com duas seções sísmicas. [13].

A análise de fácies sísmicas consiste na identificação de parâmetros de reflexão sísmica, com base principalmente em padrões de configuração que indicam fatores geológicos como litologia, estratificação e sistemas deposicionais [48]. Na lista a seguir, são descritas as facies sísmicas de cada intervalo de horizonte em termos de amplitude e continuidade.

- H1 - Este horizonte limita o topo da sucessão evaporítica rica em Permiano do Grupo Zechstein. Os reflexos superiores apresentam fácies sísmicas estratificadas, enquanto os inferiores apresentam fácies sísmicas homogêneas, onde não há reflexões aparentes.
- H2-H1: Os reflexos entre H2 e H1 são contínuos e têm baixa amplitude. Essas características estão relacionadas ao ambiente nerítico constituído principalmente por depósitos argilosos.
- H3-H2: O ambiente deposicional e a litologia dos sedimentos entre H3 e H2 são essencialmente os mesmos que em H2, mas essa porção é marcada por falhas poligonais.

Falhas poligonais são comuns em sedimentos argilosos e atribuem uma característica de montículo descontínuo aos reflexos.

- H4-H3: Os reflexos entre H4 e H3 são predominantemente subparalelos e têm amplitude variável.
- H5-H4: Caracterizado por reflexos que apresentam uma configuração sigmoide progradante de baixa energia e amplitude média a baixa devido a um sistema de delta complexo depositado durante o final do Mioceno e Plioceno.
- H6-H5: O depósito entre H6 e H5 é composto principalmente por reflexos paralelos de alta amplitude devido a depósitos de carbonatos do Grupo *Chalk* do Cretáceo Superior.
- H7-H6: Compreende o Supergrupo do Mar do Norte, que apresenta alternância de arenitos e argilitos que conferem a essas fácies as características de semicontinuidade e baixa amplitude.
- H8-H7: Este intervalo ainda compreende sedimentos do Supergrupo do Mar do Norte e apresenta fácies contornadas em forma de montículo e de baixa amplitude.
- H9-H8: As fácies entre H9 e H8 são ruidosas, possivelmente devido a ruído de aquisição ou falha no processamento sísmico.

Este banco de dados possui no total de 1602 imagens sísmicas, onde 651 imagens sísmicas são *inlines* e 951 imagens sísmicas são *crosslines*, de dimensões de 951x462, respectivamente, e cada imagem possui seu rótulo correspondente, que será utilizado posteriormente para a realização do treinamento das redes neurais.

2.3.2 Banco de dados Penobscot

Este outro conjunto de dados vem do artigo [14] e é resultado de uma pesquisa sísmica 3D pública conhecida como Penobscot 3D. Assim como no caso do conjunto de dados Holanda F3, eles também fornecem dados sísmicos rotulados. Eles identificaram sete camadas (H1, H2, ..., H7), numeradas da superfície mais profunda para a mais alta, como mostra a Figura 15. Eles dividiram o cubo sísmico em sete setores, com características geológicas semelhantes entre as camadas.

- H1: A unidade de fácies é composta por reflexos de alta amplitude. Embora a maioria dos reflexos sejam contínuos, alguns apresentam ângulos de mergulho e outros são truncados, evidenciando um ambiente de alta energia.
- H2-H1: O pacote consiste principalmente de reflexos paralelos de alta amplitude.

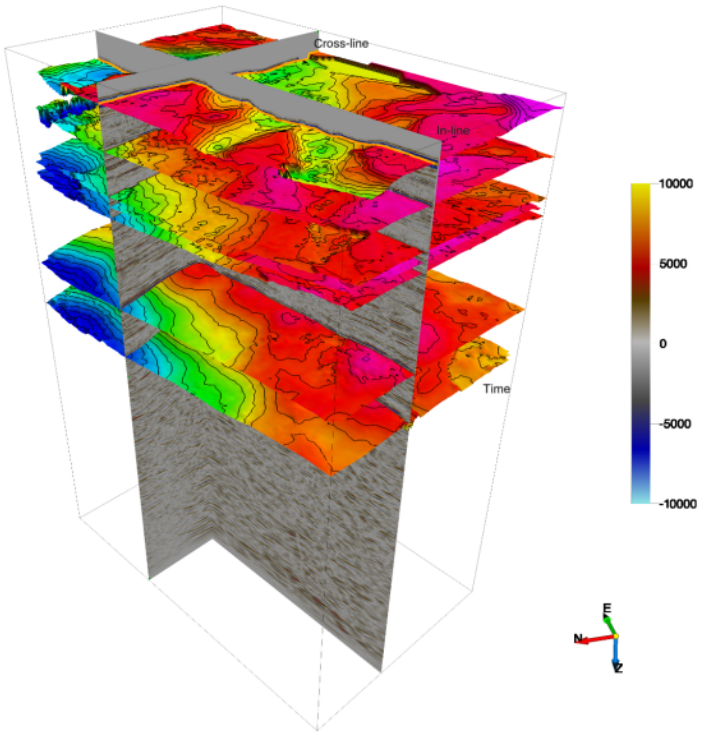


Figura 15 – Sete horizontes interpretados mostrados junto com duas seções sísmicas. [14].

- H3-H2: Os reflexos são predominantemente subparalelos e apresentam amplitude variável.
- H4-H3: Os reflexos entre esses horizontes são contínuos, mas apresentam baixa amplitude, o que dificulta sua identificação.
- H5-H4: Unidade de fácies contendo reflexos paralelos a subparalelos, como o intervalo anterior, mas menos contínuos.
- H6-H5: A unidade de fácies é caracterizada por reflexos paralelos a subparalelos, contínuos e de alta amplitude.
- H7-H6: A unidade de fácies abaixo do H7 horizonte é caracterizada principalmente por reflexos paralelos, concordantes e de alta amplitude.

Este banco de dados possui no total de 1082 imagens sísmicas, onde 601 imagens sísmicas são *inlines* e 481 imagens sísmicas são *crosslines* e cada imagem possui seu rótulo correspondente, com dimensão de 601x1501 e 481x1501 pixels, respectivamente.

3 REDES NEURAIS

Conceituar de forma correta Inteligência Artificial (IA) não é algo simples, pertence à área de computação que procura reproduzir, por meios computacionais, atividades humanas como raciocinar, planejar, aprender e resolver problemas. De forma bem resumida, a IA é a tentativa de reproduzir em uma máquina a capacidade humana de ser inteligente [49]. Dentre as várias abordagens da inteligência artificial, merecem destaque as Redes Neurais Artificiais (ANN) e as Redes Neurais Convolucionais (CNNs).

Uma das áreas de redes neurais mais recentes que vem ganhando atenção da comunidade científica é a de aprendizado profundo [50]. A aprendizagem profunda, também conhecida como *deep learning*, é uma área da inteligência artificial que utiliza redes neurais artificiais para aprender e realizar tarefas complexas. Essa abordagem revolucionária tem aplicação em diversas áreas, como reconhecimento de voz, visão computacional e processamento de linguagem natural, abrindo caminho para a automação de tarefas complexas e criação de soluções inovadoras [19].

Em [51], [52] e [53] argumentam que os métodos tradicionais de aprendizado de máquina possuem limitações no processamento de dados, principalmente relacionados à necessidade de segmentação e desenvolvimento de extratores de atributos para representar as imagens e servir de entrada para os classificadores.

Com o passar do tempo, os pesquisadores passaram a desenvolver algoritmos que integrassem os processos de extração de atributos e classificação de imagens dentro da própria rede neural [54]. Dessa maneira, na técnica de aprendizado profundo, é minimizada a necessidade de pré-processamento ou segmentação. No entanto, a aprendizagem profunda também apresenta algumas limitações. Uma das principais é a necessidade de um grande volume de dados para treinamento, geralmente na ordem de centenas a milhares de amostras. Além disso, a qualidade desses dados é crucial, pois informações incorretas ou viés nos conjuntos de treinamento podem resultar em modelos tendenciosos ou imprecisos. Outra desvantagem é a falta de transparência e interpretabilidade dos modelos de aprendizagem profunda, tornando-os uma "caixa-preta" em relação à lógica interna e aos processos de tomada de decisão. Isso pode ser problemático em áreas sensíveis, onde é importante entender o raciocínio por trás das predições e garantir a confiança e a ética na aplicação dos algoritmos [19].

O método de aprendizado profundo que mais ganhou notoriedade é a rede neural convolucional (CNN). Uma CNN é composta basicamente por três tipos de camadas: a primeira camada (convolucional) detecta e extrai os atributos; a segunda camada (*pooling*) seleciona e reduz a quantidade de atributos; e a terceira camada (totalmente conectada)

serve para integrar todas as características extraídas pelas camadas anteriores, normalmente utilizando uma rede neural do tipo Perceptron Multi-Camadas (MLP) para realizar a classificação final da imagem.

Como mostra a Figura 16, a área de Inteligência Artificial possui diversas subáreas e que a área mais estudada é a área de Redes Neurais Convolucionais. Como essa área possui diversos segmentos, o foco desse trabalho se dará nas CNNs aplicadas para classificação e reconhecimento de imagens.

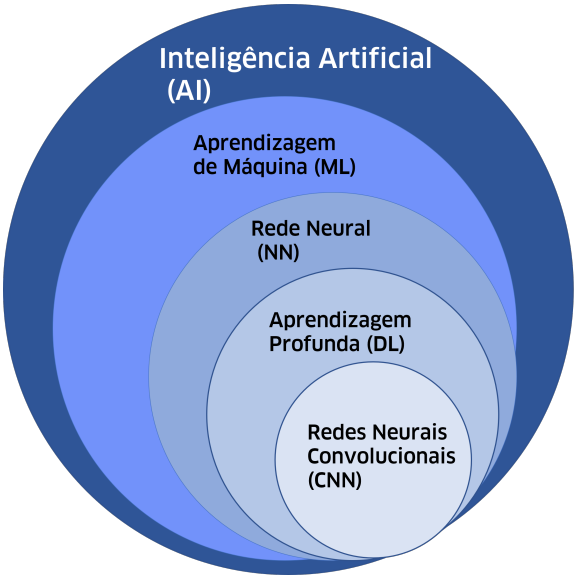


Figura 16 – As subdivisões da área de inteligência artificial.

3.1 Redes Neurais Convolucionais

Em 1980, Fukushima [55] apresentou uma rede neural artificial baseada em uma estrutura hierárquica proposta em 1962 para modelar um sistema visual. A ideia principal era aplicar neurônios com os mesmos parâmetros em diferentes partes da camada anterior, para alcançar a invariância translacional, que seria a possibilidade de captar todas as informações mais importantes em uma imagem independente da posição e rotação do objeto [56]. Essa invariância translacional é crucial para tarefas como reconhecimento de imagens, pois é importante identificar um objeto independentemente de sua posição [56]. Fukushima treinou sua rede, chamada *Neocognitron*, usando um esquema de aprendizado não supervisionado [55, 57]. Anos mais tarde, LeCun e outros desenvolveram a rede neural convolucional (CNN), seguindo as ideias do *Neocognitron*, mas usando *backpropagation* para treinamento [57], o que resultou em melhores desempenhos em tarefas de reconhecimento de padrões [58].

As redes neurais convolucionais (CNNs) são uma classe de modelos de aprendizado de máquina especialmente projetadas para processar dados na forma de matrizes, como

imagens, espectrogramas de áudio ou imagens volumétricas. Elas são compostas por camadas de neurônios interconectados, que realizam operações matemáticas específicas, como a operação de convolução, para extrair características relevantes do conjunto de dados de entrada. A estrutura da CNN aproveita as propriedades desses tipos de sinais, como a estrutura espacial das imagens ou a estrutura temporal do áudio, para realizar tarefas de processamento de imagem e reconhecimento de fala com alta precisão. Além disso, devido à sua capacidade de aprender características hierárquicas, as CNNs se mostraram eficazes em tarefas de visão computacional como classificação de imagem, detecção de objetos e segmentação de imagem.

Assim como o nome sugere, as CNNs empregam a operação linear correlação cruzada. A convolução de uma função x(t) com uma função w(t) é definida na Equação 3.1.

$$s(t) \;=\; \int_{-\infty}^{\infty} x(a)w(t-a)da \;=\; (x \;*\; w)(t) \tag{3.1}$$

Como exemplo ilustrativo, imagine que *x(t)* é a posição de uma partícula no tempo t e que o sensor para medir essa posição é ruidoso. É possível calcular a média de vários pontos de dados para obter uma versão mais limpa do sinal *x(t)*. Esse processo é conhecido como filtragem de média. Supondo que as observações recentes sejam mais relevantes do que as antigas, pode-se usar uma função de ponderação *w(a)*, onde a é a idade das medidas. Essa técnica é conhecida como filtragem exponencial. A função de ponderação é geralmente escolhida de modo a dar mais peso às observações mais recentes e menos peso às observações antigas. Aplicando essa média ponderada, obtem-se uma estimativa suavizada *s(t)* de *x(t)*. Essa operação também pode ser realizada para valores discretos de t, basta substituir a integral por uma soma infinita [59]. Esta técnica tem aplicação em diversos campos como sinais de processamento, controle e estimação.

No contexto das CNNs, x é a entrada, w é o *kernel* ou filtro e s é o mapa de recursos. A entrada geralmente é um *array* multidimensional de dados e o *kernel* é um *array* de parâmetros que serão ajustados pelo algoritmo de aprendizado. A operação de convolução é aplicada a esses *arrays*, onde o *kernel* é deslizado sobre a entrada, realizando multiplicações elemento a elemento entre a entrada e o *kernel* e somando esses resultados. Essa operação gera um novo *array*, conhecido como mapa de recursos, que contém características relevantes da entrada. Por exemplo, uma entrada pode ser uma representação gráfica bidimensional, identificada como I, e consequentemente, o *kernel* K deve ser igualmente bidimensional. A operação de combinação linear será então realizada de maneira simultânea em ambos os eixos ortogonais, tal como evidenciado pela Equação 3.2.

$$S(i,j) \;=\; (K * I)(i,j) \;=\; \sum_{m}\sum_{n} I(m,n)K(i-m, j-n), \tag{3.2}$$

onde o somatório infinito pode ser condensado para um somatório finito, assumindo que a entrada é zero em qualquer lugar fora da área da imagem. A Figura 17 ilustra um exemplo gráfico da operação de convolução aplicada em uma imagem bidimensional I. A saída neste exemplo é calculada apenas para as posições em que o kernel está totalmente dentro da imagem. Isso é chamado de convolução válida.

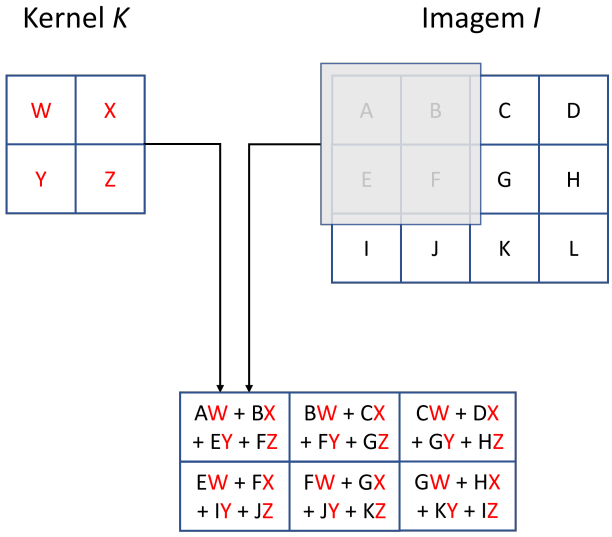


Figura 17 – Exemplo de convolução 2D.

A utilização de operações de Convolução nas redes neurais oferece vantagens relacionadas a três ideias principais: interações esparsas, compartilhamento de parâmetros e representações equivariantes [59].

Em redes neurais tradicionais, tal como o *Multilayer Perceptron* (MLP), todos os elementos da entrada interagem com os elementos da saída. Por outro lado, em CNNs, se o *kernel* de Convolução for menor que a saída, haverá conexões esparsas. Essa ideia pode ser visualizada na Figura 17: a entrada B, por exemplo, influencia somente a primeira e a segunda saídas no mapa de recursos. A conectividade escassa diminui a quantidade de memória necessária para armazenar parâmetros e o número de operações para calcular a saída. Essa redução traduz-se numa melhoria significativa de eficiência [59].

A segunda vantagem está associada ao uso conjunto de cada componente do *kernel* em todas as posições da imagem de entrada, ou seja, os parâmetros são compartilhados, como ilustrado na Figura 17. Isso significa que o mesmo conjunto de pesos é utilizado para todas as posições da imagem de entrada, o que resulta em uma diminuição significativa no número de parâmetros a serem aprendidos e aumento na eficiência em comparação com redes densamente conectadas.

Esse tipo específico de distribuição de parâmetros em uma CNN leva a uma propriedade conhecida como invariante por translação. Uma função $f(x)$ é equivalente a uma função $g(x)$ se $f(g(x)) = g(f(x))$. Se g é uma função que desloca a entrada, então

a convolução é equivalente a g. Isso significa que se movemos um objeto na imagem de entrada, por exemplo, sua representação se move da mesma forma na imagem de saída. É importante destacar que a operação de Convolução não é equivalente a transformações como mudanças de escala ou rotações.

A Figura 18 mostra um exemplo de como as camadas de uma CNN são organizadas. As camadas não lineares são omitidas nesta imagem e a área cinza representa as camadas de convolução. As camadas geralmente são divididas em três estágios: o primeiro é responsável por realizar operações de convolução, produzindo um conjunto de ativações lineares [59]. Para criar uma camada de convolução, precisa-se especificar o tamanho do *kernel*$(m \times n)$, o *stride (s)* usado para varrer a entrada e o número de mapas de recursos *(f)* gerados pela camada. Isso significa que serão utilizados f mapas de características diferentes de tamanho $m \times n$ e stride s para deslizar pelo tensor de entrada, gerando f mapas de saída. Neste trabalho, será chamada uma camada de convolução de tamanho $k \times k$, *stride* s e f mapas de características de *Conv(k, s, f)*.

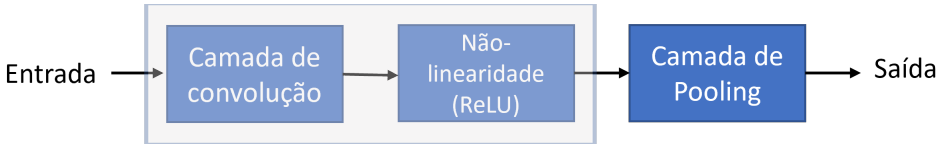


Figura 18 – Representação simplificada de um conjunto comum de camadas presentes em uma CNN.

No segundo estágio de uma rede neural de processamento de imagem, conhecido como estágio do detector, é aplicada uma função de ativação não linear às saídas da camada de convolução. A função de ativação não linear mais comumente utilizada nesta etapa é a Unidade Linear Retificada (ReLU), ela é a mais comum pois é menos complexa computacionalmente e produz excelentes resultados. A finalidade desta função é introduzir não linearidade na rede, permitindo que ela possa aprender relações não lineares entre as características das imagens e as classes a serem preditas. A Figura 19 mostra algumas funções de ativação não lineares, em (A) tem-se a função ReLU, que é descrita pela equação $f(x) = max(0, x)$, em (B) a função Sigmoide que é descrita pela equação $f(x) = \frac{1}{1+e^{-x}}$ e em (C) a Tangente Hiperbólica (Tanh) que é descrita pela função $f(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$. É importante mencionar que essa função de ativação não linear é frequentemente representada implicitamente na camada de convolução, em vez de ser desenhada separadamente.

A camada de *pooling* estabelece uma janela retangular de tamanho $k \times k$ que desliza pelo dado de entrada com um *stride* s. Os valores dentro dessa janela são multiplicados por uma função matemática, como o valor máximo (MaxPool) ou o valor médio (AvgPool). O uso de k e s geralmente são definidos como maior que 1 para diminuir a amostragem da entrada [56, 59]. Após o deslizamento da janela de $k \times k$ e espaçadas em s pixels (*stride*), o tamanho do mapa de recursos é reduzido, como mostra a Figura 20.

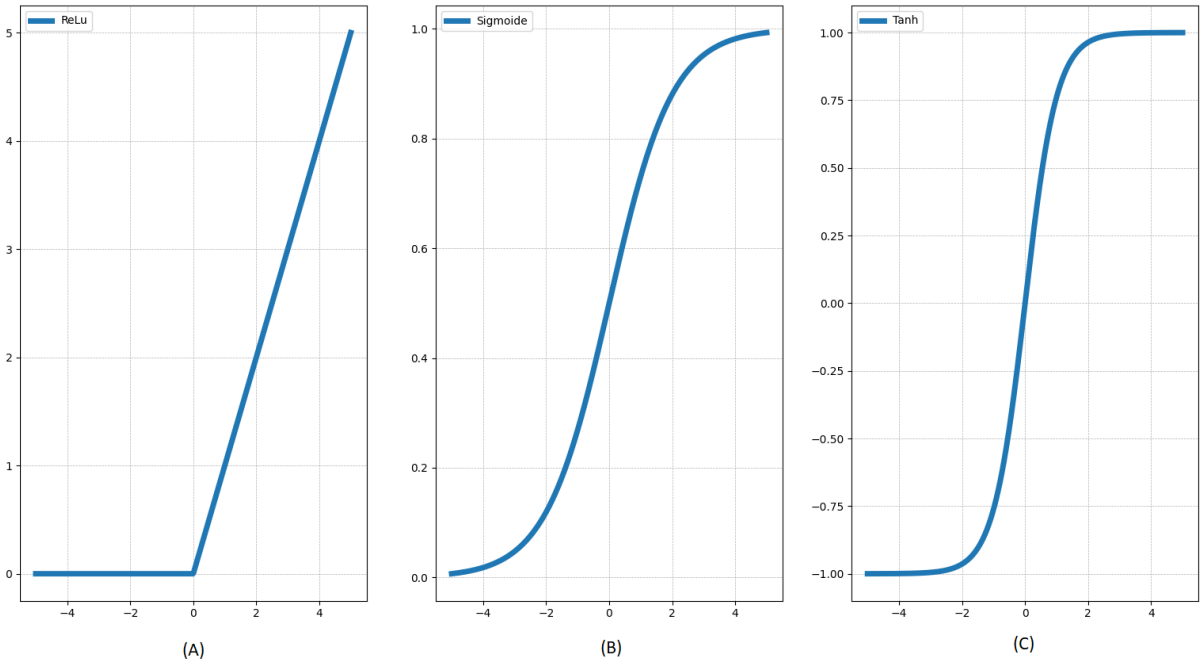


Figura 19 – Exemplo de funções de ativações, em (A) a função ReLU, em (B) a função Sigmoide e em (C) a Tangente Hiperbólica

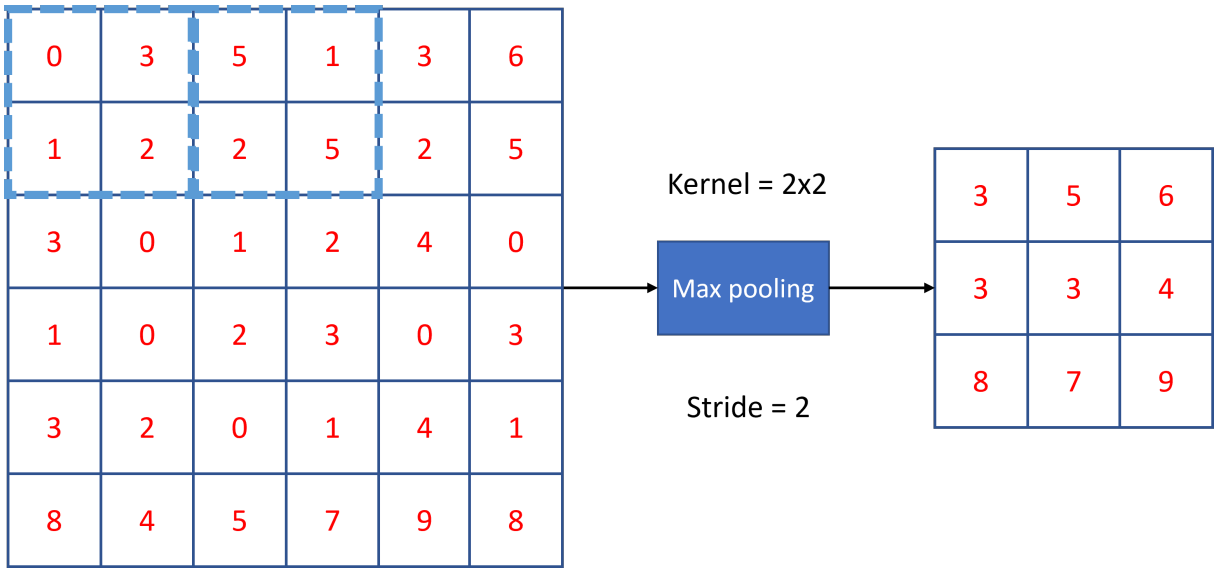


Figura 20 – Exemplo de um novo mapa de características a partir de um filtro *max pooling* de 2x2 com *stride* = 2.

Essa redução é valiosa para melhorar a eficiência, especialmente nos casos em que a invariância sob translação é mais importante do que as localizações precisas, ela se torna robusta para pequenas translações da entrada, pois as equações estatísticas permanecem estáveis na vizinhança [56, 59]. Para problemas de classificação, funções de *pooling* podem ser vantajosas, pois normalmente se está interessado apenas na presença de um determinado objeto na imagem. No entanto, ao substituir os pixels por modelos de estatísticas, pode-se perder informações de posição da característica, o que é essencial em

aplicações como a detecção de objetos. Isso corrobora o fato de que as camadas *pooling* nem sempre estão presentes em redes convolucionais, pois depende da aplicação, como visto em [38].

Uma CNN é formada pelo acúmulo repetido das camadas mencionadas anteriormente. Em uma estrutura comum, duas ou três etapas de camadas de convolução e *pooling* são inseridas, seguidas por mais etapas de camadas de convolução e camadas completamente conectadas (FC) [56]. No contexto de tarefas de classificação, a última camada é responsável por produzir pontuações para cada categoria, e geralmente é uma camada completamente conectada. Se existem 10 classes no conjunto de dados, a última camada completamente conectada deverá ter 10 unidades de neurônio.

A Figura 21 mostra a arquitetura AlexNet, que utiliza uma estrutura de CNN com 8 camadas, que surpreendentemente venceu o concurso de reconhecimento visual ImageNet 2012, essa arquitetura foi capaz de distinguir 1000 classes com uma taxa de erro de apenas 15,3%. A estrutura de AlexNet é composta por cinco camadas de processamento de convolução, duas camadas ocultas intermediárias completamente conectadas e uma camada final completamente conectada. Essa rede marcou um ponto importante, mostrando pela primeira vez que os recursos obtidos pelo aprendizado de máquina pode superar os recursos projetados manualmente, revolucionando o campo da visão computacional. É observado que a arquitetura possui camadas convolucionais seguidas por camadas de *pooling*, mas sua forma geral corresponde ao diagrama de bloco que foi demostrado anteriormente.

3.1.1 Treinamento de Redes Neurais

Após definir a forma geral de um modelo CNN, é importante descrever também o procedimento de treinamento, ou como os pesos da rede são ajustados no decorrer das interações. O aprendizado supervisionado é a tarefa de aprender uma determinada função, que mapeia dados de entrada e saída com base em exemplos de dados de entrada e saída [60]. O conjunto desses dados é chamado de conjunto de treinamento e o processo de aprendizado de uma função é chamado de treinamento. O objetivo é aproximar a função de mapeamento tão bem que, quando houver novos dados de entrada não vistos, possa prever as variáveis de saída para esses dados de maneira razoável.

A Figura 22 mostra o diagrama em blocos de um sistema utilizando aprendizado supervisionado em uma rede neural. Nesse diagrama, (X_i, Y_i) é a amostra de treino supervisionado, onde X é a entrada do sistema, Y é a saída do sistema, ou seja, a *label* da entrada X, e o sub-índice "i" é o índice que indica a amostra do treino. Durante esse processo, uma entrada de treinamento X_i alimenta a Rede de Treinamento, assim, a rede de treinamento gera uma saída $\tilde{Y}_i$. Então, essa saída é comparada a *label* original Y_i através de um subtrator, que compara a diferença entre elas. Onde essa diferença é o "erro", como mostra o diagrama da Figura 22. Este erro, é enviado à Rede de Treinamento para o ajuste

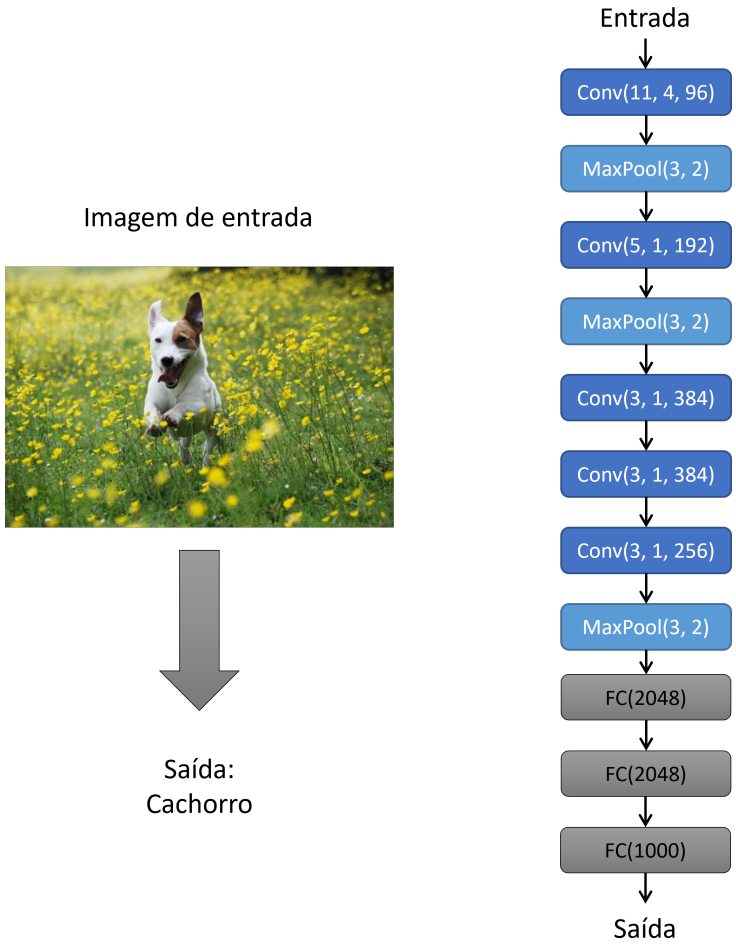


Figura 21 – A arquitetura Alexnet.

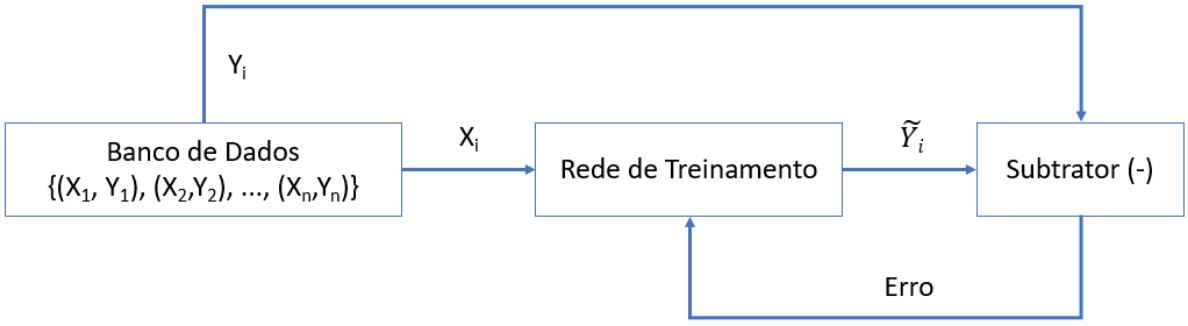


Figura 22 – Diagrama em blocos de um sistema utilizando aprendizado supervisionado.

dos parâmetros de aprendizagem.

Com isso, o intuito dessa Rede de Treinamento é obter um conjunto de parâmetros que possa minimizar a diferença entre $\tilde{Y}_i$ e Y_i, para todo "i", minimizando o erro total considerando o banco de dados de treinamento. Com isso, no final, deseja-se ter um classificador que consiga separar todas as classes treinadas, a Figura 23 mostra uma

classificação utilizando 2 classes.

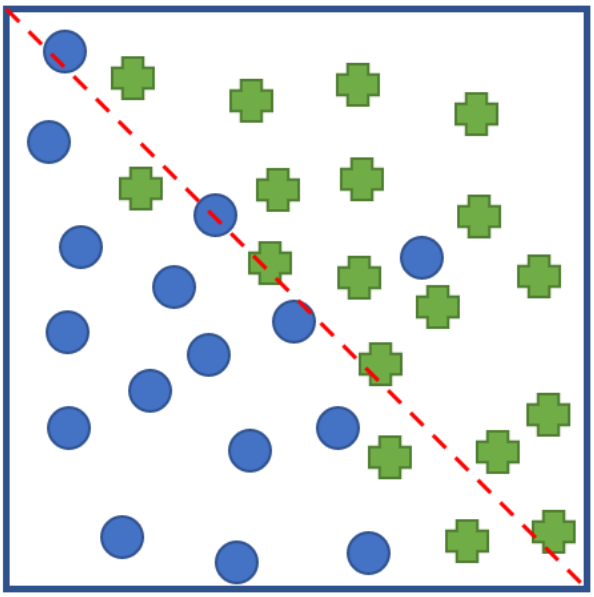


Figura 23 – Classificador de 2 classes.

Idealmente, após aprender com a coleta de dados de treinamento, o modelo será capaz de prever as classes de imagens não vistas - capacidade de generalizar o que acabou de aprender. No contexto da rede neural, observar a imagem de entrada consiste em fornecê-la à rede para que ela se propague pelas camadas. Uma pontuação para cada categoria é gerada na saída, completando a propagação direta.

O algoritmo de aprendizado consiste em minimizar essa função de erro (ou função de custo) com relação aos pesos da rede. O *Stochastic Gradient Descent* (SGD) e suas variantes são os algoritmos de otimização mais populares para treinamento de CNNs. Esses algoritmos funcionam realizando pequenas correções nos pesos da rede a cada iteração, a fim de minimizar a função de erro. O processo é repetido várias vezes, até que a função de erro alcance um valor aceitável ou não haja mais melhorias significativas.

Uma variação do SGD convencional é a inclusão de *momentum*. O SGD com momento acumula uma média móvel exponencialmente decrescente dos gradientes anteriores e continua seguindo essa direção. Um parâmetro α controla a velocidade com que as contribuições de gradientes passados irão diminuir [59].

A taxa de aprendizado é um fator importante para o algoritmo SGD, pois pode afetar significativamente o desempenho do modelo. A cada iteração do SGD, uma amostra (ou mini-batch) é selecionada aleatoriamente a partir do conjunto de dados de treinamento. A amostra pode consistir em um único exemplo (caso do SGD "puro") ou em um pequeno conjunto de exemplos (caso do mini-batch SGD). Esses algoritmos utilizam uma taxa de aprendizado separada para cada peso e adaptam essas taxas automaticamente durante o treinamento. O RMSProp é um exemplo conhecido desses algoritmos: ele reduz a taxa de

aprendizado de acordo com o histórico dos gradientes quadrados. Ele também descarta contribuições do passado extremo com a ajuda de uma média móvel exponencialmente decrescente. O RMSProp adiciona a taxa de decaimento da média móvel à lista de parâmetros a serem especificados pelo usuário [59], e seu funcionamento é descrito pela Equação 3.3.

$$\begin{aligned} M_t^2 &= \varphi M_{t-1}^2 + (1-\varphi)\nabla_t^2 \\ \Delta\theta_t &= -\nu\Delta\theta_{t-1} - \frac{\eta}{\sqrt{M_t^2+\epsilon}}\nabla, \end{aligned} \tag{3.3}$$

onde ∇ é o gradiente da função custo em relação ao θ, M^2 é a média móvel do quadrado dos gradientes, ν é o momento, ϵ é usado para estabilidade numérica no *TensorFlow*, e φ atua de forma semelhante a um termo de momento na média de decaimento exponencial.

O gradiente descendente é um algoritmo de otimização utilizado para encontrar o mínimo de uma função $F(x)$. Ele funciona movendo-se em pequenos passos no sentido oposto do gradiente da função, que é o vetor de suas derivadas parciais. Com cada passo, a função é reduzida até alcançar um mínimo global ou local. É importante escolher cuidadosamente os hiperparâmetros, como o passo de aprendizado, para garantir a convergência e eficácia do algoritmo. O processo de interação para a otimização desse algoritmo propõe um novo ponto x' que é descrito pela Equação 3.4 [59].

$$x' = x - \lambda\nabla_x f(x), \tag{3.4}$$

onde λ, a taxa de aprendizado, é um número real e positivo cujo sua função é determinar o tamanho do passo de aprendizagem. Pode-se dizer que o algoritmo convergiu quando todos os elementos do gradientes são zero [59].

3.1.1.1 Algoritmo de Retropropagação

Para calcular os gradientes da função de erro em relação a cada peso da rede, é necessário usar o algoritmo de retropropagação. Este algoritmo utiliza a regra da cadeia das derivadas de forma repetitiva, partindo da saída da rede e se movendo até a entrada. A criação desse algoritmo possibilitou a resolução do problema de atualização dos pesos nas camadas escondidas. Esse algoritmo é separado em duas fases, a propagação e a retropropagação.

- **Propagação**

  Durante esta etapa, um certo conjunto de dados padrão é aplicado à camada de entrada da CNN, e a resposta é propagada para a camada seguinte. Isso é repetido ao longo das camadas da rede, produzindo um conjunto de saída final como a resposta atual da rede. Durante essa fase, os pesos da rede permanecem fixos. É importante observar que o sinal flui através da rede da esquerda para a direita. Esses sinais que se propagam diretamente, neurônio a neurônio, através da rede desde a entrada até

a saída, são chamados de Sinais Funcionais, essa denominação surge do fato de que esses sinais são obtidos na saída de cada neurônio como uma função dos sinais de entrada. A Figura 24 ilustra a fase de propagação em uma rede MLP (4-3-2) [59].

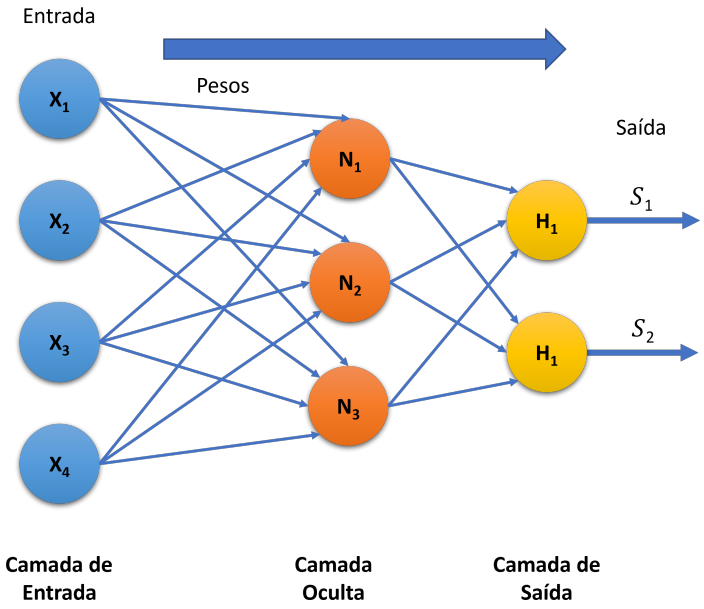


Figura 24 – Descrição da fase de propagação.

- **Retropropagação**

  A fase de Retropropagação consiste em ajustar os pesos sinápticos para minimizar o erro, começando pela camada de saída e terminando na camada de entrada. O erro é calculado como a diferença entre a resposta desejada e a resposta atual, e é propagado de volta através da rede no sentido contrário ao sinal de entrada, dando origem ao nome "retropropagação de erro".

  Ajustar os pesos da rede neural artificial (ANN) é crucial para que a resposta obtida se aproxime da desejada. Isso é feito através da retropropagação, onde o erro é calculado na camada de saída e propagado de volta através da rede, camada a camada, esse erro é chamado de sinal de erro. A Figura 25 ilustra essa fase [59].

3.1.2 Regularização

No procedimento de treinamento que foi mencionado, são calculados os erros em um conjunto de entrada chamado conjunto de treinamento e minimizado à função de custo. No entanto, qualquer modelo de aprendizado de máquina deve ter a capacidade de generalizar, ou seja, ter um bom desempenho não apenas no conjunto de treinamento, mas também em dados não vistos anteriormente [59]. Várias estratégias foram desenvolvidas para reduzir o erro de generalização, conhecidas como técnicas de regularização [59]. Serão descrito brevemente os métodos relevantes para este trabalho. Uma das técnicas de regularização

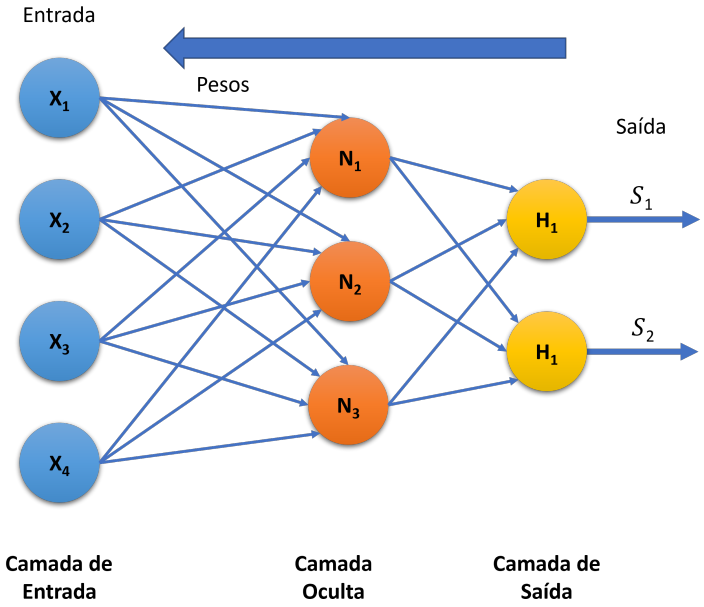


Figura 25 – Descrição da fase de retropropagação.

mais comuns é a penalidade pela norma do parâmetro L^2 ou decaimento dos pesos. A ideia é adicionar um termo de penalidade à função de custo para favorecer os menores pesos quadrados de norma L^2. Considerando a função de custo $C(w; X; y)$, a expressão para o decaimento dos pesos terá a forma descrita conforme a Equação 3.5 [59].

$$C_{reg}(w; X, y) \;=\; C(w; X, y) \;+\; \frac{\alpha}{2}\;||w||^2, \tag{3.5}$$

onde $\alpha \in [0, \infty)$ é um hiperparametro que controla a contribuição dos parâmetros da penalidade relacionada a função de custo original. Esse resultado visa minimizar o $C_{reg}(w; X, y)$, selecionando pesos que fazem um trade-off entre ajustar os dados de treinamento e ter valores pequenos [59].

Embora a perda de peso seja fácil de usar, pode-se argumentar que uma maneira eficaz de generalizar melhor um modelo é treiná-lo com mais dados. No entanto, tem-se acesso a um conjunto limitado de exemplos. Para alguns problemas, pode ser fácil criar dados fictícios para aumentar o conjunto de treinamento, com um método chamado de *data augmentation* [61].

Na tarefa proposta de classificação de objetos, é fácil simular variações reais aplicando operações simples nas imagens. Por exemplo, pode-se preencher imagens e cortá-las aleatoriamente; pequenas mudanças no brilho também são possíveis; giros aleatórios são escolhas comuns. Além disso, o aumento de dados pode ser realizado durante o treinamento. Ao apresentar um exemplo de treinamento para a rede, ela tem 50% de chance de ser modificada por uma dessas operações. Dessa forma, a rede pode se tornar mais resistente a essas variações sem precisar coletar mais dados.

Outra abordagem bem-sucedida e mais recente é chamada de normalização de lotes (*Batch Normalization - BN*) [62]. Embora projetada principalmente para melhorar a convergência em modelos profundos, a normalização em lote também pode atuar como um regularizador [59, 62, 63].

Os modelos de aprendizagem profunda consistem em muitas camadas e as entradas para uma camada são influenciadas pelas anteriores. Ao treinar tais modelos com SGD, atualiza-se os pesos de acordo com os gradientes [59]. No entanto, pequenas mudanças nos pesos podem ser amplificadas com o aumento da profundidade da rede neural [62]. Assim, após a atualização, as entradas de cada camada podem ser um pouco modificadas, o que pode gerar um atraso na convergência, pois as camadas devem se adaptar constantemente a novas distribuições de entrada. A ideia do BN é reduzir a variância dessas distribuições. Por exemplo, considerando uma ativação x na rede e um mini-batch B, têm-se m valores para esta ativação: $B = \{x_1, ..., x_m\}$. A transformada do BN é definida conforme a Equação 3.6 e Equação 3.7 [62].

$$BN_{\gamma,\beta}(x_i) \;=\; \gamma\hat{x}_i + \beta; \qquad \hat{x}_i \;=\; \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + \epsilon}}; \tag{3.6}$$

$$\mu_\beta \;=\; \frac{1}{m}\sum x_i \;; \qquad \sigma_\beta^2 \;=\; \sum (x_i \;-\; \mu_\beta)^2, \tag{3.7}$$

onde μ_β e σ_β^2 são as médias dos mini-batch e variância, μ e β são parâmetros de aprendizagem, e σ é a constante para estabilidade numérica [62]. Essa normalização introduz ruído aditivo e multiplicativo nas camadas, o que pode ter um efeito de regularização [59].

É importante observar que os métodos apresentados podem ser usados em conjunto, pode-se aplicar a normalização de lotes e *data augmentation* simultaneamente, por exemplo.

3.1.3 Aprendizagem Residual

A normalização em lote é uma técnica que ajuda na convergência do treinamento de redes neurais, estabilizando os gradientes e aumentando sua convergência. Com isso, foi criada uma nova ferramenta para que os pesquisadores conseguissem trabalhar com redes cada vez mais profundas. No entanto, com a profundidade da rede, surge o problema da degradação da precisão, onde a precisão satura e depois se deteriora rapidamente e aumenta o erro de treinamento da rede [64].

Para resolver este problema, foi proposto o uso do aprendizado residual. Neste método, a ideia é comparar duas redes: uma rede superficial e outra profunda, criada adicionando camadas de identidade na rede menor. O objetivo é que a rede profunda não apresente erros de treinamento maiores que a rede superficial, mas como os experimentos

mostram o contrário, o método atual é incapaz de encontrar soluções melhores ou mesmo semelhantes ao método da identidade [64].

Como solução, em [64] os autores propõem que as camadas sejam adicionadas em um mapeamento residual em vez do mapeamento original não referenciado. Desta forma, o objetivo passa a ser zerar o resíduo em vez de tentar ajustar diretamente a identidade. Se as camadas de identidade empilhadas fossem a solução ótima, seria mais fácil zerar o resíduo do que tentar ajustar diretamente a identidade.

Definindo-se o mapeamento subjacente como $H(x)$, o mapeamento residual então se torna $F(x) = H(x) - x$. O mapeamento original se torna $F(x) + x$. A Figura 26 (a) mostra uma unidade residual que implementa a nova formulação de $H(x)$. Então, em vez de usar camadas convolucionais convencionais, a rede residual é construída empilhando unidades específicas.

O atalho (ou *shortcut*) na unidade da Figura 26 (a) pode ser usado diretamente se a entrada x e a saída $F(x) + x$ tiverem as mesmas dimensões. Quando os tamanhos não correspondem, há duas abordagens típicas para corrigir esse problema. O primeiro preenche a entrada com zeros ao realizar a soma. A segunda usa um atalho de projeção, representado na Figura 26 (b), que aplica uma convolução 1x1 para fixar as dimensões.

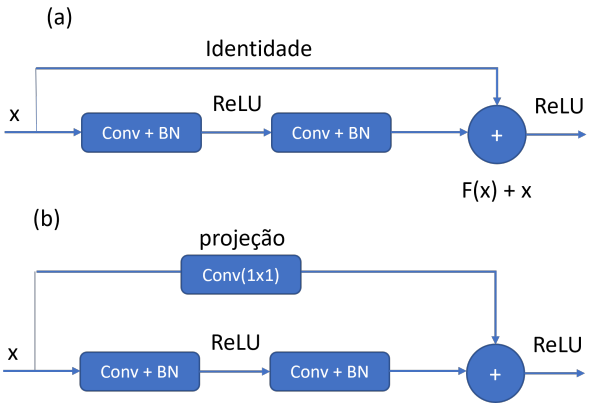


Figura 26 – (a) Unidade residual com atalho de identidade. (b) Unidade residual com atalho de projeção

3.2 Segmentação Semântica

A interpretação de informações visuais tem sido abordada de diversas maneiras ao longo dos anos, mas o processo permanece fundamentalmente o mesmo: examinar imagens com o objetivo de identificar objetos e avaliar sua relevância [65]. O problema de aprendizado a partir de informações visuais é geralmente categorizado em várias tarefas, incluindo classificação de imagem [66], localização e detecção de objetos [64], segmentação semântica e segmentação de instâncias [67], entre outras [68]. Essas tarefas desempenham

um papel crucial em áreas como visão computacional, inteligência artificial e robótica, e têm sido objeto de intensa pesquisa na última década, impulsionada pela disponibilidade de grandes conjuntos de dados rotulados e pelo aumento no poder computacional.

A segmentação de imagens é um dos principais processos constituintes de um sistema de Visão Artificial. Seu objetivo é separar as regiões que compõem uma imagem para que seja possível a extração de atributos (características) desta imagem conforme mostra a Figura 27. Regiões espacialmente diferentes, em uma mesma imagem, são caracterizadas como áreas com nível de cinza semelhante, ou mesma textura, ou outra característica em comum. Portanto, a segmentação consiste em atribuir um rótulo a cada pixel da imagem, identificando-o a qual região ele pertence, de acordo com suas características comuns [69].

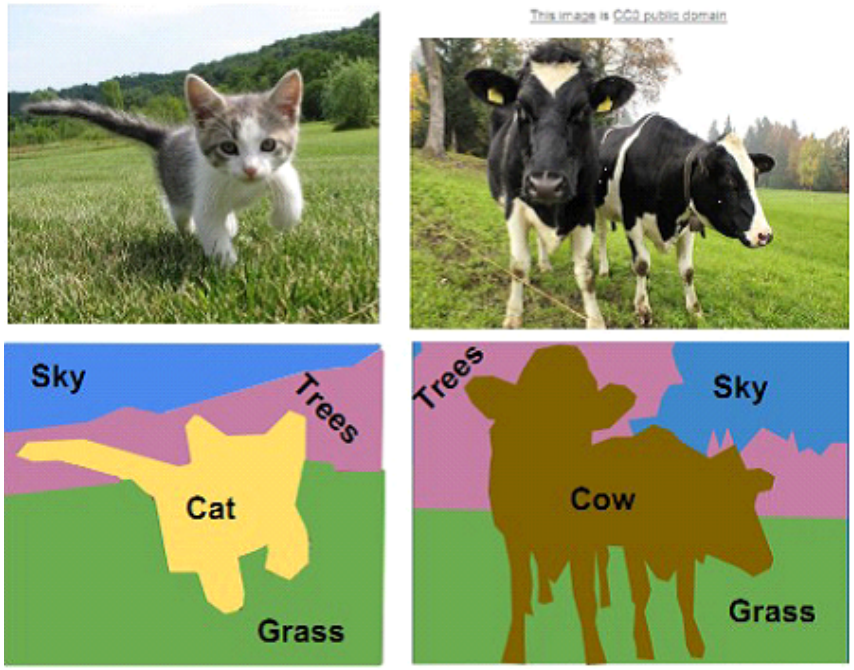


Figura 27 – Segmentação semântica [15].

As técnicas de segmentação, como limiar e crescimento de regiões, não se comportam bem na solução do problema de segmentação de imagens de microestruturas, dada a dificuldade de achar o limiar adequado ou as sementes adequadas para o correto crescimento das regiões. É necessária a aplicação de filtros no pré-processamento da imagem em análise, embora nem sempre é uma metodologia que resolva. Para amostras diferentes do mesmo material, é utilizada uma adaptação dos filtros ou a escolha de outros. Com base nestes problemas de segmentação utilizando-se algoritmos tradicionais de processamento digital de imagens, surge a necessidade do emprego de um método mais eficiente nesta aplicação que são as ANN [70].

A segmentação de imagens é um processo crítico na visão computacional, o qual envolve a divisão de uma entrada em segmentos para simplificar a sua análise. O problema fundamental na segmentação é particionar uma imagem em regiões que satisfaçam a

algum critério. Essas regiões representam objetos ou partes dos mesmos e compreendem conjuntos de pixels ou super-pixels, os quais são rotulados na forma de algum critério pré-estabelecido [71]. Essa segmentação é a rotulação de cada pixel da imagem com um rótulo específico para cada tipo de objeto, se na mesma imagem existir mais de um objeto do mesmo tipo, todos serão identificados a partir do mesmo rótulo [72].

3.3 U-Net

A abordagem de uma *Fully Convolutional Network*, foi introduzida por Long, Shelhamere e Darrell [17, 35], e propuseram uma adaptação de redes já existentes, que são utilizadas para classificação de imagens, transformando as camadas totalmente conectadas em camadas de convolução. Isto permite a geração de mapas de características de segmentação para cada imagem, podendo ser de qualquer tamanho.

- *Encoder*: Os mapas de características gerados possuem resolução espacial bem menores do que a imagem original. Este processo de redução da resolução espacial da imagem de entrada através de uma série de convoluções ocorre no caminho chamado de caminho de contração, ou, também, chamado de *encoder*;
- *Decoder*: Na saída do *encoder*, é feito um aumento da resolução espacial através de uma série de deconvoluções dos mapas de características para a resolução inicial da imagem de entrada. O caminho onde este processo de expansão da resolução espacial ocorre é chamado de *decoder* ou caminho de expansão [17].

.

Pode se definir a U-Net como uma versão estendida das redes totalmente convolucionais (FCN), de forma a proporcionar segmentações mais precisas com conjuntos de treinos pequenos. A principal diferença em relação à FCN se faz na camada de *decoder*, onde há um número grande de mapas de características, tendo como consequência uma simetria em relação à camada de *encoder*, assim tomando um formato de "U", como mostra a Figura 28. A forma adotada para aumentar o número de mapas de características no *decoder* é concatenar a saída de uma camada de *unpooling* com a saída da camada de convolução do *encoder* correspondente. Desta forma, a rede aprende o conteúdo da informação da imagem e onde este conteúdo está localizado na imagem.

Em relação à tarefa de segmentação, é necessário que a rede neural seja capaz de combinar a informação da localização com a informação contextual dos mapas de características da imagem a ser predita. A U-Net é capaz de juntar as informações contextuais obtidas do caminho de contração (*encoder*) com as informações de localização

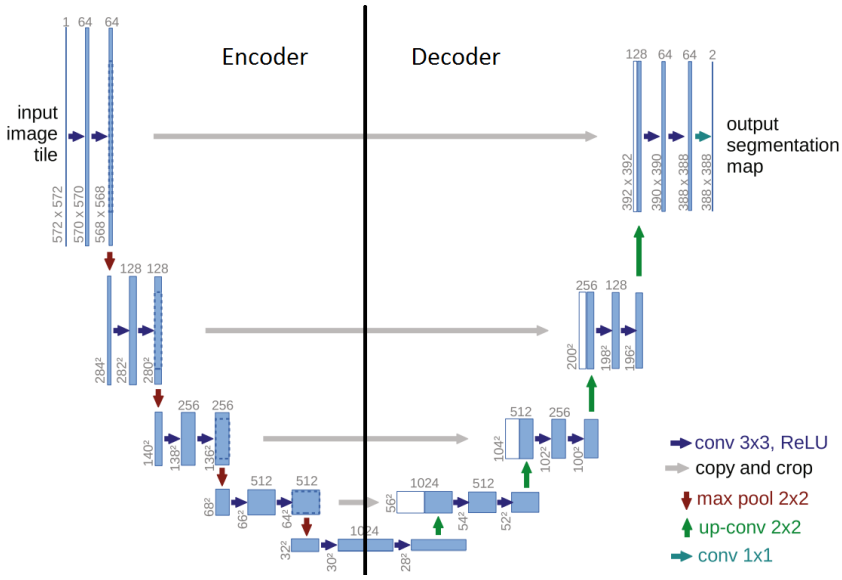


Figura 28 – Arquitetura de uma rede *Encoder* – *Decoder* [16].

adquiridas do caminho de expansão (*decoder*) com um bom desempenho. Sem precisar necessariamente do uso de pré ou pós processamento [35].

A utilização da U-Net com outros classificadores para a tarefa de segmentação demonstra que o campo de pesquisa de aprendizagem profunda tem avanços contínuos, com diferentes tipos de aplicações [73].

3.4 DANET-FCN3

A arquitetura Danet-FCN3 foi baseada na topologia VGG-FCN [17], onde os autores conseguiram converter uma camada totalmente conectada no final da rede em convoluções de 1x1. Essa arquitetura utiliza blocos residuais para extrair as características, que seria o *encoder*, e uma unidade transposta da mesma estrutura para recriar a imagem original com todas as suas *labels*, que seria o *decoder*.

A Figura 29 mostra todas as camadas da arquitetura Danet-FCN3. Onde o S é o *stride*, #CH é o número de canais (filtros) utilizado em todas as operações dentro do bloco. Assim, a primeira entrada C5 S2 64, significa uma convolução com 64 filtros de tamanho 5x5 e *stride* de 2. Res e Rest são compostos por 3 blocos de unidade residual e unidade residual transposta, respectivamente. O bloco Res possui uma unidade residual com *stride* igual a 2, seguido de duas unidades residuais com o *stride* igual a 1. O bloco Rest possui duas unidades residuais transpostas com *stride* igual a 1, seguido de uma unidade residual transposta com o *stride* igual a 2. Ru e Rut são as unidades residual e residual transposta, respectivamente, como é mostrado na Figura 29. Cada unidade residual é composta de uma convolução seguida de *Batch Normalization* (Conv+BN), de ativação ReLU, de Conv+BN somado à entrada (*Shortcut*) e uma ativação ReLU. A unidade residual transposta segue a

mesma arquitetura, porém, com a convolução transposta. Os parâmetros utilizados podem ser encontrados em [38].

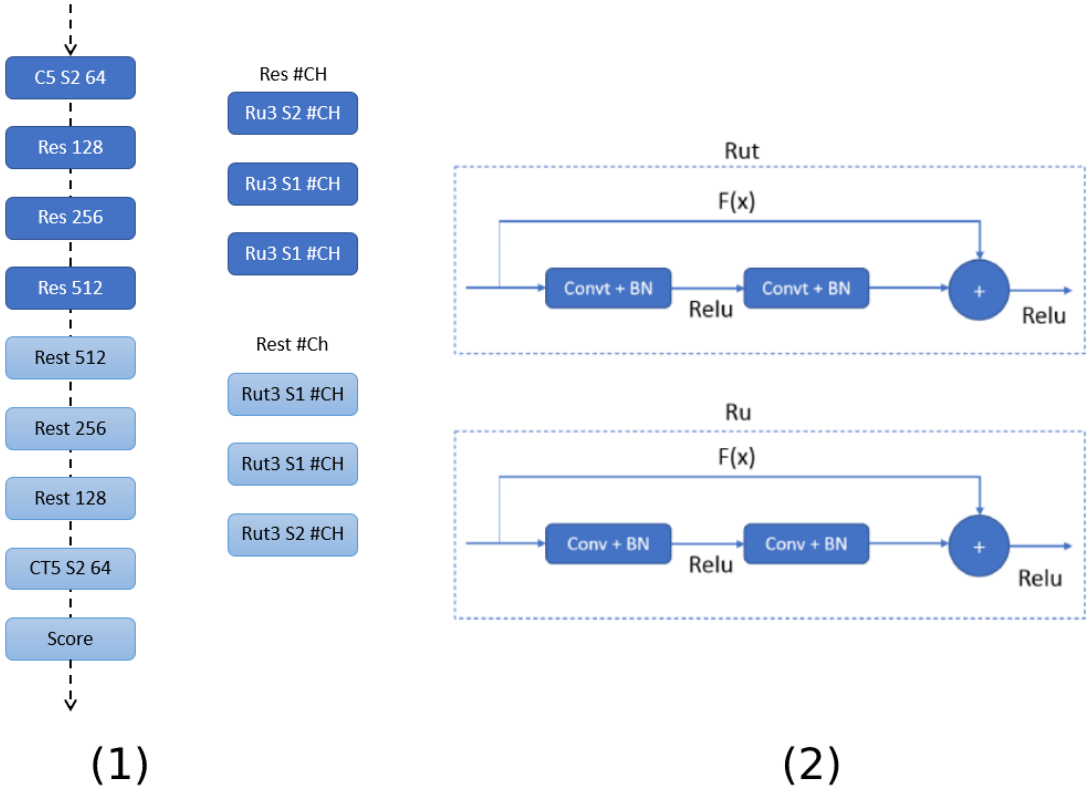


Figura 29 – (1) - Topologia do modelo Danet-FCN3. Na direita está a descrição de cada bloco [17]. (2) O bloco Ru é a unidade residual e o bloco Rut é a unidade residual transposta [18].

4 Preparação do banco de dados

A arquitetura Danet-FCN3 é mais complexa que a U-Net em situações em que o banco de dados de treino é reduzido. Contudo, ela possuiu um desempenho superior, isto é, um $mIoU$ superior [32]. Dada a menor complexidade, a U-Net torna-se atrativa. Porém, vê-se a necessidade de melhorar seu desempenho. Assim, visando melhorar o desempenho dessa rede é proposto um pré-processamento, baseado na filtragem adaptativa nas imagens sísmicas.

Como levantado em [74] e [75], pode-se identificar diferentes tipos de substratos utilizando as características das texturas das imagens sísmicas. Com isso, pode-se automatizar o processo utilizando um modelo de rede neural para que se consiga distinguir as texturas nas imagens sísmicas.

O primeiro passo foi utilizar um filtro na imagem, o *Contrast Limited Adaptive Histrogram Equalization* (CLAHE), para aumentar o contraste dos pixels da imagem.

4.1 Equalização de Histograma

O aprimoramento de imagem pode ser alcançado por meio de diferentes métodos, e a equalização de histograma é uma técnica relativamente simples. Vários trabalhos ilustram a melhoria significativa na qualidade das imagens com pouca iluminação, graças à equalização de histograma. Para obter exemplos típicos dessa técnica, pode-se consultar [76] e [77]. O procedimento comum consiste em remapear as escalas de cinza da imagem, de forma que o histograma resultante se aproxime de uma distribuição uniforme. No entanto, essa abordagem pressupõe uniformidade na qualidade da imagem em todas as áreas, e que um único mapeamento em escala de cinza fornece aprimoramento semelhante para todas as regiões da imagem. Quando a distribuição de tons de cinza varia de uma região para outra, essa suposição não é válida. Nesse caso, uma técnica de equalização de histograma adaptativo pode ser empregada para superar a abordagem padrão de maneira significativa.

A principal ideia por trás da equalização do histograma adaptativo é encontrar um mapeamento para cada pixel com base em sua distribuição local de escala de cinza (vizinhança). Esse método, desenvolvido independentemente em [78] e [79], aplica um mapeamento de contraste aprimorado em um pixel específico, utilizando os valores de intensidade imediatamente ao redor do pixel. No entanto, esse processo precisa ser repetido um número de vezes igual ao número de pixels na imagem, resultando em um alto custo computacional que não é viável para aprimoramento de imagens em tempo real.

Uma abordagem que equilibra a equalização de histograma global e o algoritmo

adaptativo completo é a equalização do histograma regional. Nessa técnica, a imagem é dividida em um número limitado de regiões, e a mesma técnica de equalização do histograma é aplicada a pixels em cada região. Embora esse método possa ser otimizado para processamento em tempo real, ele apresenta uma falha importante. Devido às diferenças entre os mapeamentos de duas regiões vizinhas, os pixels nos dois lados do limite entre as regiões serão mapeados de maneira diferente, resultando em um efeito de bloqueio significativo.

Uma opção ao algoritmo totalmente adaptável consiste em estimar o mapeamento necessário para cada pixel por meio da escolha de um pequeno conjunto de regiões contextuais na imagem e pelo cálculo do mapeamento para um determinado pixel como uma interpolação bilinear dos mapeamentos derivados de regiões próximas. Em outras palavras, o mapeamento para um pixel é obtido pela soma ponderada dos mapeamentos de suas quatro regiões mais próximas, sendo os pesos calculados de acordo com a proximidade do pixel aos centros dessas regiões. Uma descrição detalhada dessa técnica pode ser encontrada em [80] e [81].

Em algumas situações, quando a distribuição de escala de cinza é altamente localizada, não é desejável transformar imagens de baixo contraste usando a equalização total de histograma, uma vez que a curva de mapeamento pode incluir segmentos com inclinações acentuadas, fazendo com que duas escalas de cinza próximas sejam mapeadas para escalas de cinza significativamente diferentes. Esse problema é resolvido por meio da limitação do contraste permitido pela equalização de histograma. A combinação dessa abordagem de limitação de contraste com a equalização do histograma adaptável mencionada acima resulta no que é conhecido como Equalização de Histograma Adaptativo Limitada por Contraste (CLAHE), discutida em [82] e resumida em [83]. A formulação completa dessa abordagem é brevemente revisada na próxima seção. No entanto, antes de iniciá-la, será realizada uma breve revisão de alguns dos problemas de implementação relacionados à equalização adaptativa de histograma.

Uma implementação eficiente do algoritmo CLAHE é descrito em [84]. Neste artigo, é apresentado o projeto de uma máquina de processamento paralelo de equalização adaptativa de histograma, que é usada para implementação do algoritmo CLAHE em imagens médicas estáticas. Com essa implementação, os pesquisadores conseguiram reduzir significativamente o tempo de processamento com sua tecnologia atual. Para uma imagem de 512 x 512, o tempo de processamento foi reduzido de 1 a 2 horas para 10 segundos. Esse tempo de processamento é considerado razoável para o processamento de imagens médicas estáticas. Outro método de processamento paralelo para implementação da equalização do histograma adaptativo é relatado em [85]. Neste trabalho, é relatada uma aceleração de 6,5: 1 para uma máquina de 8 nós e uma aceleração de 16,5: 1 para uma máquina de 26 nós. Ambos os métodos são custosos e são adequados apenas para imagens estáticas.

4.2 Equalização de Histograma Adaptativo Limitada por Contraste (CLAHE)

A equalização de histograma adaptativo limitada por contraste tem sido bem-sucedida na melhoria de imagens médicas, um grande exemplo é [86] que foi realizado um pré-processamento em uma imagem de ressonância magnética, tornando-se melhor para avalizações. Com seu grande sucesso nessa área médica, está sendo proposta a utilização da mesma técnica em imagens sísmicas, melhorando seu contraste para que se consiga ter melhores resultados durante o aprendizado da rede.

Esse método é formulado por meio da divisão da imagem em várias regiões quase iguais e não sobrepostas. Para imagens de tamanho 512 x 512, um bom número de regiões é geralmente 64, com a imagem dividida em 8 partes iguais em cada direção. Um exemplo dessa divisão pode ser visto na Figura 30. Essa divisão resulta em três grupos distintos de regiões. O primeiro grupo é composto apenas por quatro regiões e é conhecido como a classe de regiões de canto (RC). O segundo grupo, composto por 24 regiões, é chamado de classe de regiões de borda (RB). Todas as regiões na borda da imagem, exceto as regiões de canto, pertencem a essa classe. O último grupo, composto por 36 regiões, é denominado classe de regiões internas (RI).

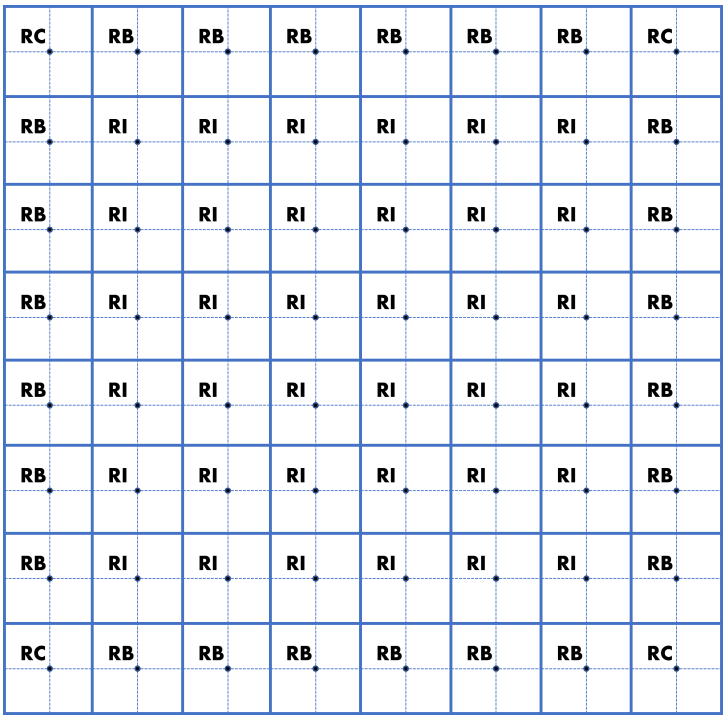


Figura 30 – Exemplo da divisão da imagem pelo filtro de CLAHE.

Nesta estratégia, primeiramente é calculado o histograma de cada região. Em seguida, é obtido um limite de corte com base em um limite de expansão de contraste desejado. Em seguida, cada histograma é redistribuído de modo que sua altura não exceda

o limite de recorte. Finalmente, as funções distribuições acumuladas, (CDF) [87], são determinadas para o mapeamento em tons de cinza dos histogramas limitados de contraste resultantes. Na técnica CLAHE, os pixels são mapeados por meio da combinação linear dos resultados dos mapeamentos das quatro regiões mais próximas. A formulação dessa abordagem para as regiões no grupo RI é simples. Entretanto, para as regiões nos grupos RC e RB, essa formulação requer algumas considerações especiais.

4.2.1 Cálculo da Função de Mapeamento

Para cada valor de escala de cinza, a contagem de pixels correspondentes em todas as regiões é contabilizada, resultando em uma estimativa aproximada da densidade de tons de cinza na imagem. A equalização de histograma é obtida usando uma estimativa da função CDF. Se M e N representam o número de pixels e tons de cinza em cada região, horizontal e vertical, respectivamente, e $h_{i,j}(n)$, para $n = 0, 1, 2, ..., N\text{-}1$, é o histograma da região (i, j), então é possível obter uma estimativa adequada do CDF, dimensionado por $(N\text{-}1)$ para o mapeamento de tons de cinza [88], descrito na Equação 4.1.

$$\begin{aligned} f_{i,j}(n) &= \tfrac{(N-1)}{M} \sum h_{i,j}(k); \\ n &= 1,\ 2,\ 3,\ \ldots,\ N\ -1. \end{aligned} \tag{4.1}$$

Essa função pode ser utilizada para transformar a função de densidade em tons de cinza de forma aproximada em uma função de densidade uniforme, através do processo conhecido como equalização de histograma. No entanto, essa abordagem pode aumentar o contraste da região ao máximo, o que pode não ser desejado. Para limitar o contraste a um nível desejado, a inclinação máxima da função é limitada a um valor pré-determinado. Uma forma de limitar essa inclinação é definir um limite de corte para cortar todos os histogramas. Esse limite de corte pode ser expresso em termos de um fator de corte, em porcentagem [88], conforme a Equação 4.2.

$$\beta = \frac{M}{N}\left(1 + \frac{\alpha}{100}(s_{max} - 1)\right). \tag{4.2}$$

Neste caso, para fator de corte de zero por cento, $\alpha = 0$, o limite do clipe torna-se exatamente igual a (M/N), o que resulta em um mapeamento de identidade que distribui uniformemente todos os pixels regionais em todas as possíveis escalas de cinza. Nenhum pixel é alterado nesse caso. Para o limite máximo de clipe, alcançado em $\alpha = 100$, o limite é definido como o máximo de $(s_{max} * M/N)$, o que significa que a inclinação máxima permitida é s_{max}. Normalmente, s_{max} é definido como quatro para imagens estáticas. No entanto, para outras aplicações, é recomendável selecionar um bom valor para s_{max} por experimentação. À medida que o fator de corte varia de zero a cem, a inclinação máxima em cada mapeamento varia de um a s_{max}.

A alteração do histograma original, baseada no desejo de mudança de contraste da imagem, é realizada limitando o número máximo de contagens para cada escala de cinza a um determinado valor. As contagens extras acima desse limite são redistribuídas uniformemente entre as escalas de cinza com contagens menores que o limite do clipe. Essa redistribuição é feita de modo que a contagem de qualquer escala de cinza nunca ultrapasse o limite estabelecido. A simples redistribuição de contagens em cada histograma pode exigir várias iterações e o número de iterações tende a aumentar à medida que o fator de corte percentual diminui. Para cada região, o mapeamento em tons de cinza é obtido usando a Eq. (4.1) em seu histograma modificado.

4.2.2 Combinação de Funções de Mapeamento

Para as regiões pertencentes ao grupo RI, o mapeamento de cada quadrante é realizado com base nos mapeamentos das quatro regiões vizinhas mais próximas. Por exemplo, na Figura 31 (A), um pixel específico no primeiro quadrante da região (i, j) é mapeado levando em consideração as distâncias verticais e horizontais entre os centros das regiões (i, j), $(i, j-1)$, $(i-1, j)$ e $(i-1, j-1)$, como ilustrado na Figura 31 (B). Se a função de mapeamento para os pixels na região (i, j) for denotada por $f_{i,j}(\cdot)$, então o novo valor do pixel p no primeiro quadrante da região (i, j), com base nas distâncias especificadas, é calculado conforme a Equação 4.3 [88].

$$\begin{aligned} p_{new} = \frac{s}{r+s}\left(\frac{y}{x+y} f_{i-1,j-1(P_{old})} + \frac{x}{x+y} f_{i,j-1(P_{old})}\right) \\ + \frac{r}{r+s}\left(\frac{y}{x+y} f_{i-1,j(p_{old})} + \frac{x}{x+y} f_{i,j(p_{old})}\right). \end{aligned} \tag{4.3}$$

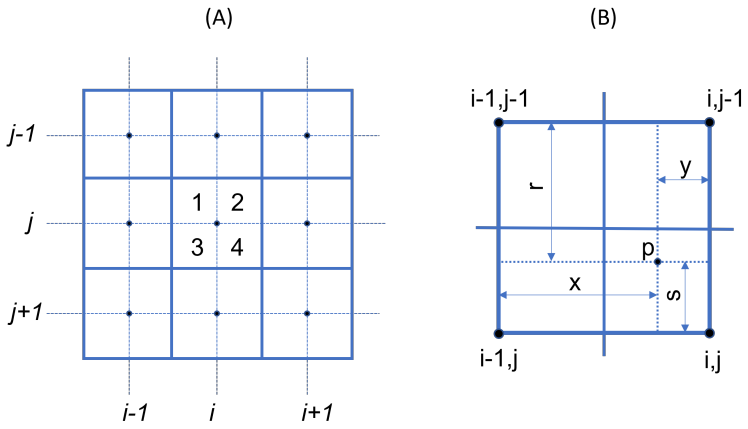


Figura 31 – Em (A) é mostrado a região RI com todos os seus vizinhos, em (B) mostra o Pixel p do quadrante 1 da região (i, j) e sua relação aos centros das quatro regiões mais próximas.

Mapeamentos semelhantes podem ser derivados para pixels em outros quadrantes da região (i, j).

Nas regiões pertencentes ao grupo RB, a configuração da vizinhança apresenta variações. A Figura 32 (A) ilustra um desses casos, no qual a estrutura da vizinhança para os pixels localizados nos quadrantes 1 ou 3 é idêntica às regiões do grupo RI. Porém, para os pixels situados nos quadrantes 2 ou 4, a estrutura da vizinhança é diferente. Por exemplo, a Figura 32 (B) mostra a configuração do quadrante 2, onde a nova escala de cinza para o pixel p é calculada através de Equação 4.4 [88].

$$p_{new} = \frac{s}{r+s} f_{i-1,j-1}(p_{old}) + \frac{r}{r+s} f_{i,j}(p_{old}). \tag{4.4}$$

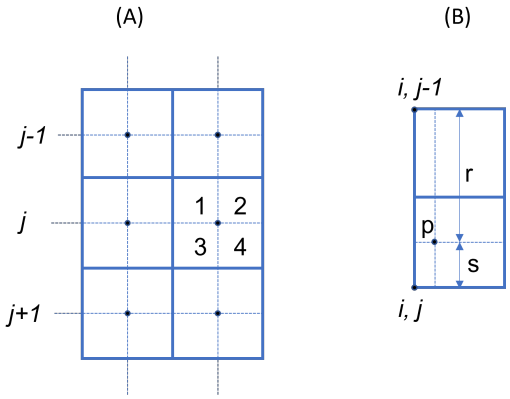


Figura 32 – Em (A) é mostrado a região RB com todos os seus vizinhos, em (B) mostra o Pixel p do quadrante 2 da região (i, j) e sua relação com o centro da região mais próxima.

Um mapeamento análogo pode ser derivado para os pixels localizados no quadrante 4 da região (i, j). Nota-se que a região (i, j) apresentada na Figura 32(A) está situada na borda direita da imagem. Todas as regiões do lado direito no grupo RB exibem comportamento semelhante. Analogamente, as regiões pertencentes ao grupo RB que estão localizadas nas bordas superior e inferior do lado esquerdo seguem o mesmo padrão. Nessas situações, os dois quadrantes que requerem um mapeamento semelhante ao do quadrante 4, conforme descrito acima, não são os mesmos ilustrados na Figura 32 (A). As regiões pertencentes ao grupo RC apresentam diferentes características em seus quadrantes. A Figura 33 ilustra um exemplo típico de uma região localizada no canto superior esquerdo desse grupo. Em relação às discussões acerca dos grupos RI e RB, observa-se que o quadrante 4 possui uma estrutura de vizinhança semelhante àquelas presentes nas regiões do grupo RI, enquanto que os quadrantes 2 e 3 exibem uma estrutura de vizinhança semelhante aos dois quadrantes laterais das regiões do grupo RB. Por sua vez, o quadrante 1 é único neste grupo e não tem contato com outras regiões. Para os pixels situados nesse

quadrante, a função de mapeamento é a mesma do mapeamento regional, sem considerar outras regiões. Nesse caso, o mapeamento seria de acordo com a Equação 4.5.

$$p_{new} = f_{i,j}(p_{old}) \tag{4.5}$$

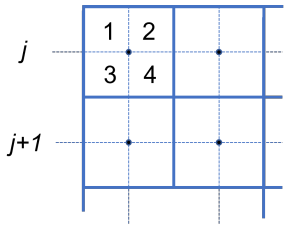


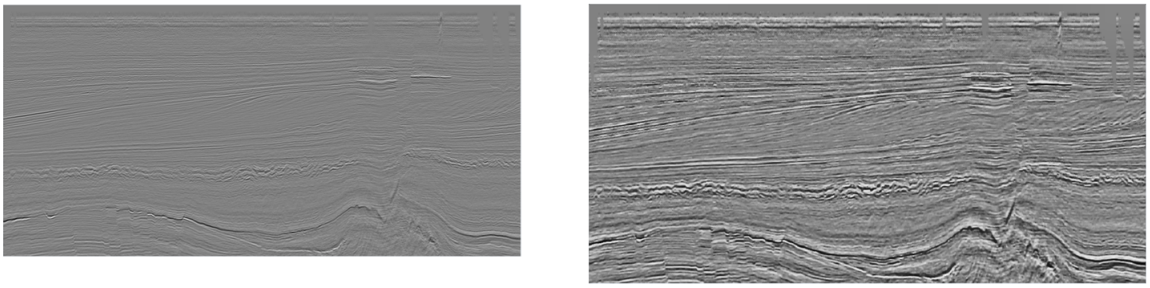
Figura 33 – Região RC com todos os seus vizinhos.

As outras três regiões pertencentes ao grupo RC apresentam comportamento semelhante. Em resumo, o processo de equalização de histograma adaptativo limitado por contraste é aplicado a cada região, baseado no fator de corte desejado. Em seguida, o mapeamento em tons de cinza de cada região é obtido através da aplicação da equação (4.1) ao histograma modificado correspondente. A imagem final é construída através da combinação dos mapeamentos equalizados de histograma adaptativo limitado por contraste das regiões cujos centros estão situados nos quatro vizinhos mais próximos, dois mais próximos ou um ou mais vizinhos mais próximos do pixel em questão. A maneira como esses mapeamentos são combinados é determinada pelas distâncias horizontais e verticais entre o pixel e os centros mais próximos, seguindo equações semelhantes ou idênticas às equações (4.3), (4.4) e (4.5).

4.3 Pré-processamento dos dados

Uma imagem sísmica *inline* do banco de dados da Holanda possui a dimensão 951x462, porém, para facilitar os cálculos foi realizado um redimensionamento, gerando uma imagem de 1024x512. Com esse novo valor, foi possível realizar alguns testes utilizando os tiles divididos em 64x64, 128x128, 256x256 e 512x512 para a aplicação do filtro CLAHE. Após análise dos resultados, optou-se por manter o valor de 256x256 devido ao seu bom desempenho. Além disso, o limite de contraste de 40 foi escolhido devido aos resultados satisfatórios obtidos, conforme ilustrado na Figura 34.

As imagens após a aplicação do filtro possuem um nível de cor de cinza entre [-30.000 , 33.000]. Com isso, foi necessário reescalar os valores em um padrão de 8 bits inteiros, pois essa etapa faz com que amplitudes extremas sejam desconsideradas, melhorando assim a representação das imagens do banco de dados.



(A) (B)

Figura 34 – Em (A) mostra a imagem original e em (B) mostra a imagem utilizando o filtro *CLAHE* e redimensionamento.

Uma abordagem bem comum é reduzir o número de classes do banco de dados devido à similaridade das classes, no banco de dados da Holanda, no final gerando apenas 6 classes, que são as Upper (H1, H2), Middle (H3, H4), Lower (H5 e H6), Rijnland/Chalk (H7), Scruff(H8) e Zechstein (H9). Com isso, dentro dos 8 bits são ajustadas todas as 6 classes e suas respectivas máscaras com os 6 rótulos.

A mesma abordagem foi realizada no banco de dados de Penobscot, foi realizada uma junção nas classes de H2, H3 e H4 devido à sua espessura muito pequena, o que dificultaria o treinamento quando utilizados *tiles* maiores.

Para trabalhar com as imagens dos bancos de dados é necessário subdividi-las em pequenos *tiles*, que é um processo amplamente utilizando, como em [74] e [75], pois, quando se utiliza imagens grandes em entradas de redes neurais o modelo se torna muito custoso e complexo computacionalmente. Com isso, foi escolhido trabalhar com *tiles* de 64x64, 128x128, 256x256 e 512x512 a partir das imagens ampliadas de 1024x512 (Holanda) e 600x1500 (Penobscot) sem *overlap*.

Para balancear o banco de dados de treinamento, foi escolhido balancear o número de amostras por classes, dando um peso diferente de acordo com a quantidade de classe para o treinamento, assim, facilitando as métricas utilizadas para medir o desempenho da rede, como Precisão ou $mIoU$. A Figura 35 mostra a porcentagem de pixels por classes.

O $mIoU$ é a Média da razão entre a área da intercessão e da área da união das classes de um conjunto de imagens. Dentro do contexto de segmentação de imagens o $mIoU$ é calculado com a saída da rede convolucional e a máscara utilizada para treino. Para manter o padrão de imagens de teste, como em [32], foram utilizadas 100 imagens sísmicas, onde 70% foi utilizado para treinamento da rede e 30% para validação em ambos os bancos de dados.

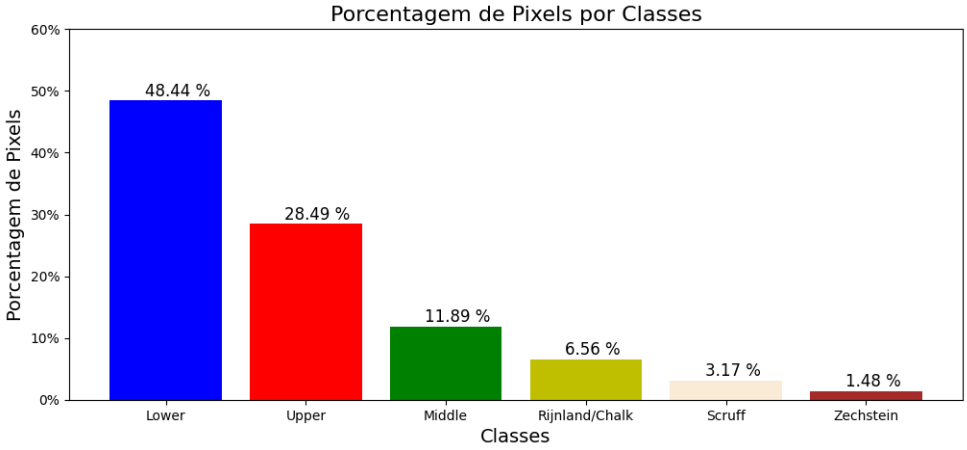


Figura 35 – Porcentagem de pixels de diferentes classes no conjunto de treinamento no banco de dados da Holanda F3.

5 Resultados e Discussões

Para a classificação das camadas rochosas, existem diversas abordagens utilizando aprendizagem profunda, como, por exemplo, as mostradas em [38] e [41]. Em ambos os artigos os autores conseguiram classificar as camadas do banco de dados da Holanda [89].

Neste trabalho, para a realização do treinamento, foi escolhido utilizar um *Batch Size* de 128, com um *mini-batch* de 64 com o número de épocas de 500. O otimizador utilizado foi o *RMSprop* por apresentar um bom desempenho, com os parâmetros de $\nu = 0.9$, $\epsilon = 1.0$ e $\varphi = 0.9$, com a função de perda *Cross Entropy* e modos de peso temporal, com os valores de coeficiente de decaimento dos pesos em 5×10^{-4}. Para o *batch normalization* foi escolhido utilizar $\beta = 1 \times 10^{-5}$ e $\mu = 0.997$. Para o *dropout* foi utilizado $\rho = 0.5$ nas camadas totalmente conectadas, e por fim, uma taxa de aprendizagem fixada em $\lambda = 0.01$.

5.1 Banco de dados - Holanda F3

A arquitetura Danet-FCN3 foi treinada com as imagens sem pré-processamento, similar ao realizado em [38], e a arquitetura U-Net utilizou o pré-processamento proposto por esta pesquisa. A Figura 36 e a Figura 37 mostram o histórico de treino da rede U-net e Danet-FCN3. Observa-se que o treino da Danet-FCN3 leva mais tempo para a convergência, necessitando mais épocas de treinamento do que a U-net. Além disso, a arquitetura Danet-FCN3 possui fortes oscilações na acurácia de validação.

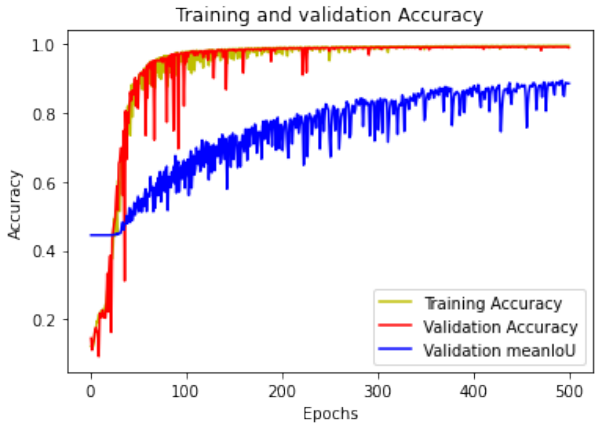


Figura 36 – Histórico das épocas de treinamento da arquitetura U-Net no banco de dados Holanda F3.

A Tabela 2 mostra os resultados da validação, com as 600 imagens, da Danet-FCN3 e U-Net no banco de dados da Holanda F3. O uso da filtragem adaptativa como pré-processamento das imagens foi benéfico para a rede U-Net de forma a superar o

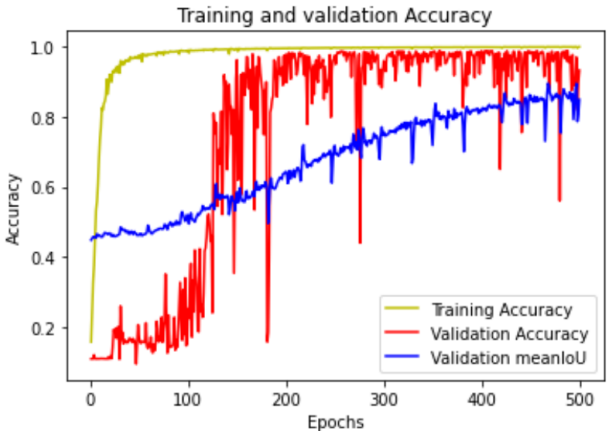


Figura 37 – Histórico das épocas de treinamento da arquitetura Danet-FCN3 no banco de dados da Holanda F3.

Tabela 2 – Valores de mIoU de acordo com o tamanho dos *tiles*.

Tamanho do Tile	U-Net	U-Net+CLAHE	Danet-FCN3
64x64	65,38%	72,46%	62,99%
128x128	83,28%	89,91%	91,41%
256x256	82,75%	90,02%	88,62%
512x512	75,35%	81,79%	78,21%

desempenho da U-Net convencional, e quase superou a Danet-FCN3 em todos tamanhos de *tiles*, nesse banco de dados.

Além disso, foi proposto verificar como o tamanho dos *titles* interferem no treinamento das arquiteturas propostas. A Figura 38 mostra os resultados obtidos. Esse levantamento foi realizado utilizando 600 imagens *inlines* que não foram utilizadas para treinamento ou validação. A métrica utilizada foi o mIoU, ou seja, das 600 imagem, cada máscara criada possuiu um valor de IoU em comparação com a máscara original, no final foi feito uma média de todas as 600 imagens testadas, gerando um mIoU, que se encontra no gráfico. Além disso foi comparado o valor mínimo de mIoU ente todas as imagens, o valor máximo de mIoU entre todas a imagem. Com isso, dentro dos valores de 64x64, 128x128, 256x256 e 512x512 foi possível verificar que o valor de *title* ideal para todos os testes variam entre 128x128 e 256x256 pixels.

A Figura 39 e a Figura 40 são as máscaras criadas pela redes após todo o treinamento e comparadas com a máscara original, criada por um geólogo. Mesmo utilizando apenas 100 imagens para o treinamento das redes, o desempenho pode ser considerado muito bom, ainda que com algumas falhas na classificação dos *tiles*, é possível distinguir todas as 6 classes da imagem sísmica. Em [90] foi demostrado que essas arquiteturas também podem ser utilizadas com mais classes, nesse artigo foram utilizado 10 classes em vez de 6, e também conseguiu alcançar um mIoU de até 90%.

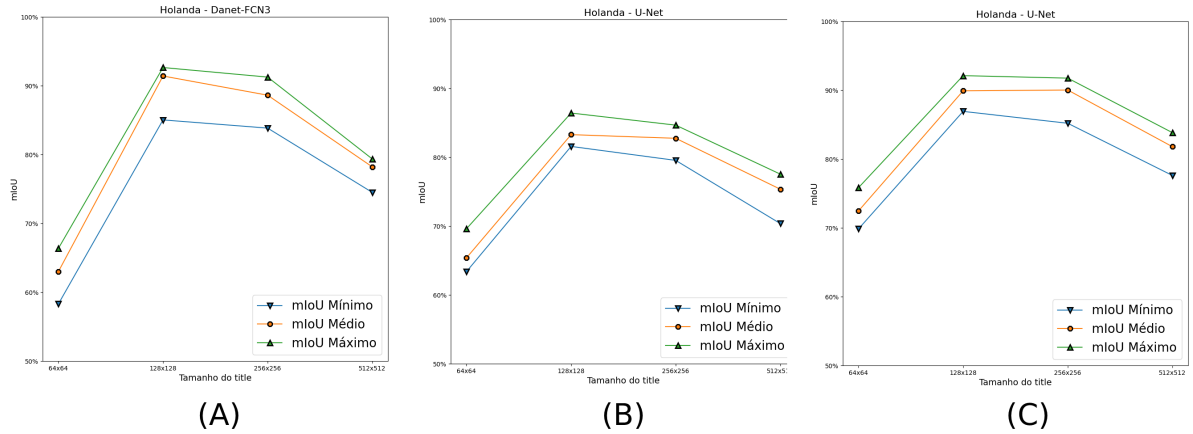


Figura 38 – Variação do *Tile* Holanda F3. (A) - Arquitetura Danet-FCN3, (B) - Arquitetura U-Net, (C) - Arquitetura U-NET+CLAHE.

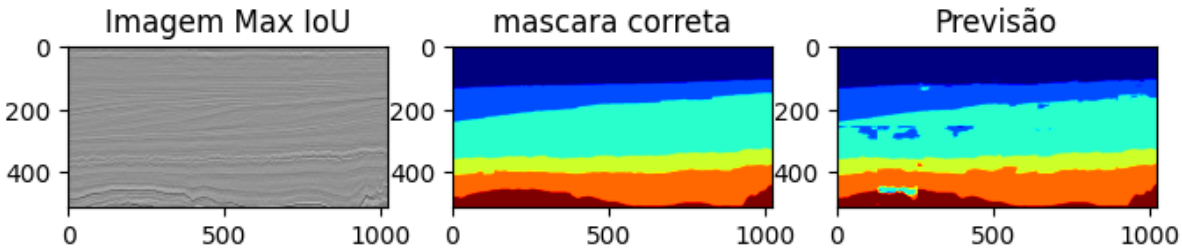


Figura 39 – Máscara criada a partir da arquitetura U-Net + CLAHE.

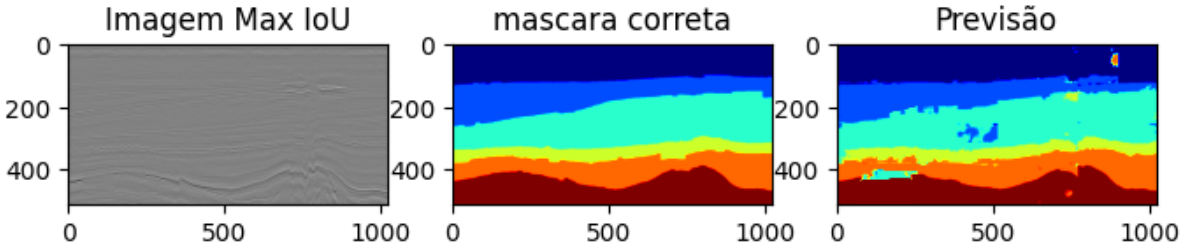


Figura 40 – Máscara criada a partir da arquitetura Danet-FCN3.

Notou-se que os erros apresentados na arquitetura Danet-FCN3 se concentraram no canto inferior esquerdo das imagens, como pode-se ver na Figura 40, enquanto os erros da U-Net concentraram-se nas primeiras camadas, entre a *Upper* e *Middle*. É necessário um estudo mais aprofundado para investigar a razão desses erros. No entanto, a hipótese mais plausível reside na influência do tamanho dos *tiles*. Devido à natureza das camadas iniciais, que apresentam menor espessura, o tamanho dos *tiles* tende a abranger múltiplas classes, o que implica em dificuldades no treinamento da rede neural.

A Tabela 3 mostra também os milhões de parâmetros e o tempo de treinamento da rede, como o filtro CLAHE não interfere diretamente nas camadas convolucionais da rede, nenhum tipo de parâmetro novo é adicionado, assim, não elevando o nível de complexidade

da rede. No final, são adicionados apenas poucos minutos de pré-processamento nas imagens sísmicas antes do treinamento da rede.

Tabela 3 – Informações importantes das arquiteturas utilizadas no banco da Holanda.

Nome do modelo	U-Net	U-Net+CLAHE	Danet-FCN3
Milhões de Parâmetros	1,94	1,94	24,91
Dias de Simulação	1,26	1,27	5,27

5.2 Banco de dados - Penobscot

Para esse banco de dados foi utilizada a mesma abordagem descrita anteriormente, entretanto, não se chegou a bons resultados na arquitetura Danet-FCN3. Foram realizadas algumas dezenas de simulações, variando diversos tipos de parâmetros, porém, não foi possível chegar nos mesmos resultados que [32] devido à alta complexidade da rede, como mostra a Tabela 3. Por isso, os resultados não serão comparados utilizando a Danet-FCN3, apenas a U-Net e U-Net + CLAHE.

A Tabela 4 mostra a variação de mIoU de acordo com o tamanho dos *tiles*, o tamanho ótimo está entre 128x128 e 256x256, tanto na U-Net quanto na U-Net + CLAHE.

Tabela 4 – Variação do mIoU de acordo com o tamanho dos *tiles* no banco de dados de Penobscot.

Tamanho do Tile	U-Net	U-Net+CLAHE
64x64	62,41%	66,84%
128x128	77,60%	80,11%
256x256	74,96%	79,52%
512x512	63,22%	65,23%

A Figura 41 mostra a variação do mIoU de acordo com o tamanho dos tiles, mostrando o mIoU máximo, mínimo e a média de mIoU. Isso demonstra que mesmo no pior dos casos, ainda é possível ter uma máscara aceitável, que consiga distinguir todas as 6 classes.

A Figura 42 mostra o histórico de treinamento da arquitetura U-Net + CLAHE, pode-se notar uma grande Volatilidade durante a validação do mIoU com o passar das épocas, é muito provável que isso se dê aos hiper-parâmetros, uma das possíveis saídas para a melhoria disso seria a diminuição da taxa de aprendizagem após a época 200, provavelmente o otimizador iria convergir mais lentamente, porém, sem tantas oscilações.

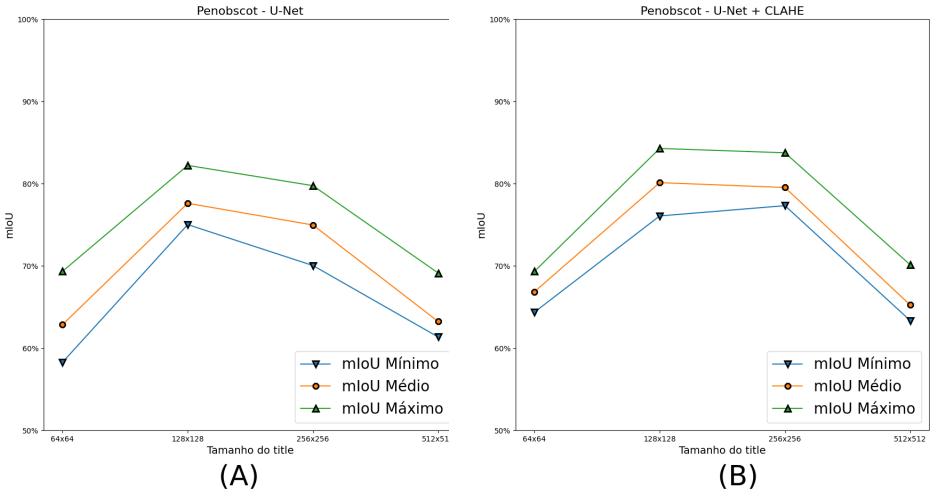


Figura 41 – Variação do tamanho dos *Tiles* no banco de dados de Penobscot. (A) - Arquitetura U-Net, (B) - Arquitetura U-Net + CLAHE.

Tabela 5 – Informações importantes das arquiteturas utilizadas no banco de Penobscot.

Nome do modelo	U-Net	U-Net+CLAHE
Milhões de Parâmetros	1,94	1,94
Dias de Simulação	1,92	1,92

A Figura 42 ilustra o registro de desempenho do treinamento da arquitetura U-Net + CLAHE. Observa-se uma notável volatilidade na métrica mIoU durante a fase de validação ao longo das épocas. Essa instabilidade provavelmente é atribuída aos hiperparâmetros utilizados. Uma estratégia viável para melhorar essa situação seria a redução da taxa de aprendizado após a época 200, o que poderia resultar em uma convergência mais lenta, porém com menor oscilação por parte do otimizador.

A Tabela 5 mostra a quantidade de parâmetros e o tempo de treinamento da rede. O filtro CLAHE não afeta as camadas convolucionais, portanto não adiciona parâmetros extras ou aumenta a complexidade da rede. Apenas alguns minutos de pré-processamento são necessários nas imagens sísmicas antes do treinamento.

A Figura 43 mostra a máscara criada pela arquitetura U-Net+CLAHE e U-Net, mesmo com alguns erros de descontinuidade, é claramente possível separar todas as camadas sem ajuda de um especialista da área.

Outro ponto importante é que o autor Civitarese em [32] não conseguiu obter bons resultados na aplicação da arquitetura U-Net nesse banco de dados, não foi mencionado o motivo. Assim, este também é mais uma contribuição para o campo de segmentação de

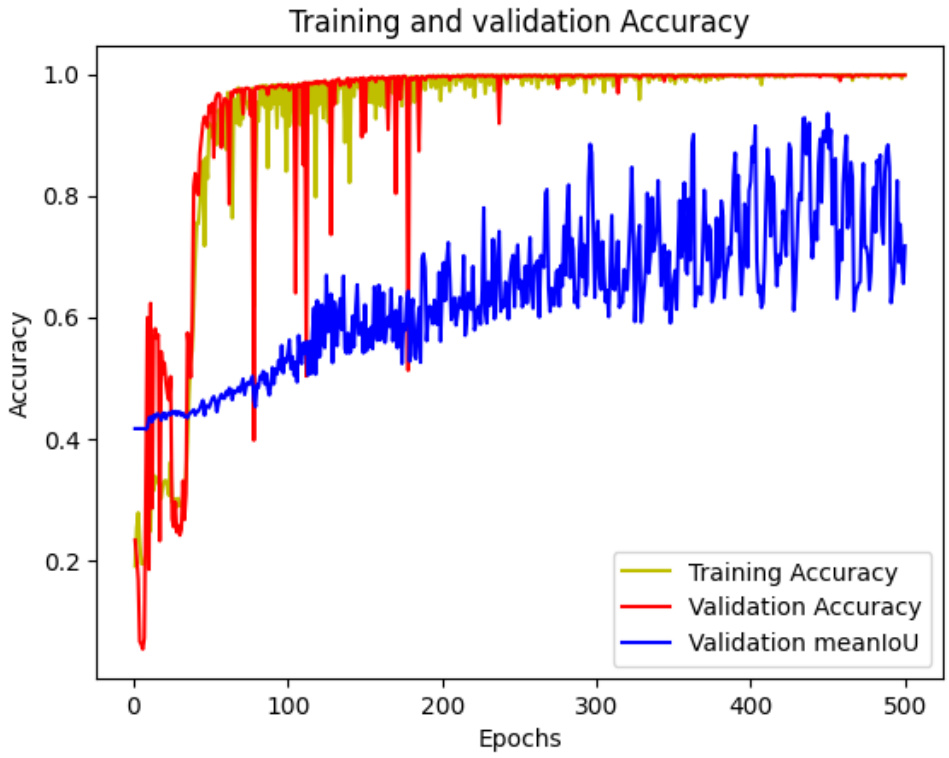


Figura 42 – Histórico das épocas de treinamento da arquitetura U-Net+CLAHE no banco de dados de Penebscot

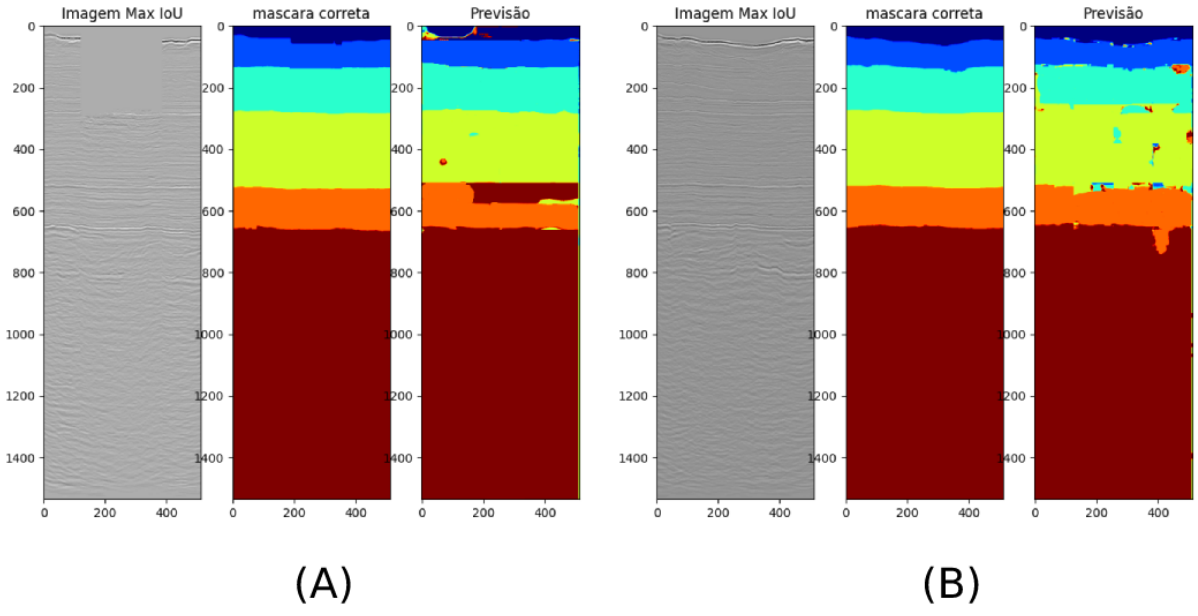


Figura 43 – Resultado das máscaras criadas pelas arquiteturas (A) - U-Net + CLAHE e (B) U-Net.

imagens sísmicas.

6 Conclusão

6.1 Considerações Finais

Neste trabalho, foi apresentada uma nova abordagem científica com o objetivo de aprimorar a extração de características nas camadas de convolução de redes profundas, por meio do pré-processamento de imagens sísmicas. A proposta consistiu na utilização do filtro de equalização adaptativa de histograma com limitação de contraste (CLAHE). Essa técnica tem sido amplamente utilizada em diversas áreas de processamento de imagens para melhorar a qualidade visual e realçar detalhes relevantes.

Verificou-se que abordagens anteriores que utilizavam segmentação semântica para dados sísmicos não exploravam o uso de pré-processamento para aprimorar a interpretação dos dados nas camadas de convolução. Portanto, decidiu-se investigar a influência do pré-processamento nesse contexto, a fim de verificar se seria possível obter resultados comparáveis ao uso de uma rede neural mais complexa, porém com uma estrutura menos exigente.

Para isso, comparou-se dois modelos de aprendizado profundo: o modelo U-Net, amplamente reconhecido por apresentar bons resultados em tarefas de segmentação, e o modelo Danet-FNC3, especificamente projetado para segmentação de imagens sísmicas. A escolha desses modelos se deve à sua eficácia comprovada em tarefas semelhantes e à relevância na literatura científica.

O modelo proposto demonstrou a capacidade de melhorar o desempenho da arquitetura U-Net sem aumentar significativamente sua complexidade, alcançando resultados excelentes e, em alguns casos, superando o desempenho do modelo Danet-FCN3. Isso indica que o pré-processamento das imagens sísmicas, por meio da aplicação do filtro CLAHE, contribuiu de forma positiva para a extração de características nas camadas de convolução. Essa descoberta é relevante, pois sugere que é possível obter resultados satisfatórios sem recorrer a estruturas mais complexas e computacionalmente intensivas.

Os experimentos foram conduzidos utilizando imagens de dois bancos de dados reais: o banco de dados da Holanda F3 e o banco de dados de Penobscot. Esses bancos de dados são amplamente utilizados na área de exploração de petróleo e gás, representando desafios reais encontrados na indústria. A utilização de dados reais fortalece a validade dos resultados obtidos e amplia a aplicabilidade da abordagem proposta.

Importante ressaltar que, em todos os treinamentos realizados, foram utilizadas apenas 100 imagens, sem a aplicação de técnicas de aumento de dados (*data augmentation*). Essa restrição visa avaliar o impacto direto do pré-processamento proposto na arquitetura

das redes neurais. No entanto, a utilização de um maior número de imagens e a aplicação de técnicas de aumento de dados poderiam potencialmente melhorar ainda mais os resultados obtidos.

Além disso, realizou-se um estudo sobre o impacto do tamanho dos *tiles* no treinamento das arquiteturas. Por meio de testes e análises estatísticas, foi investigada a variação do mIoU em função do tamanho dos *tiles*, sendo constatado que o tamanho ideal, para os bancos de dados utilizados, situa-se entre 128x128 e 256x256 pixels. Essa faixa de tamanho permitiu capturar informações contextuais suficientes para a tarefa de segmentação, sem sobrecarregar a capacidade de processamento das redes neurais. Valores acima ou abaixo dessa faixa apresentaram dificuldades para um treinamento adequado das arquiteturas, resultando em desempenho inferior. De acordo com o estudo conduzido, o tamanho recomendado que apresenta maior probabilidade de gerar um bom valor de mIoU é de 256x256 pixels.

Essas descobertas têm implicações significativas na área de processamento de imagens sísmicas, contribuindo para o desenvolvimento de técnicas mais eficazes e eficientes na interpretação desses dados. O pré-processamento das imagens por meio do filtro CLAHE pode ser considerado como uma etapa valiosa na pipeline de processamento, proporcionando ganhos expressivos de desempenho sem aumentar a complexidade das redes neurais. Além disso, a identificação do tamanho ideal de "tiles" para o treinamento das arquiteturas contribui para um uso mais eficiente dos recursos computacionais e pode orientar o projeto de novas soluções em segmentação de imagens sísmicas.

6.2 Trabalhos Futuros

Para trabalhos futuros, há várias direções promissoras a serem exploradas com base nos resultados e descobertas deste estudo. Primeiramente, recomenda-se testar a nova abordagem proposta em novos bancos de dados, além daqueles utilizados neste trabalho, a fim de avaliar a generalização e a robustez do modelo em diferentes contextos e desafios específicos de cada conjunto de dados. Isso permitiria uma análise mais abrangente do desempenho do modelo e a validação de sua eficácia em cenários diversos.

Além disso, uma possível extensão seria aumentar a complexidade do treinamento, expandindo o número de classes a serem segmentadas. Ao incorporar mais classes, o modelo poderia abordar cenários mais complexos e apresentar um nível superior de detalhamento na segmentação das imagens sísmicas. Essa expansão permitiria uma análise mais refinada das informações contidas nas imagens e um melhor entendimento das estruturas presentes nas camadas geológicas, contribuindo para a interpretação geológica de forma mais precisa.

Outro aspecto importante a ser explorado é o aumento do número de imagens disponíveis para o treinamento. A utilização de conjuntos de dados maiores, possivelmente

obtidos por meio da aquisição de dados sísmicos adicionais ou do acesso a bancos de dados mais extensos, permitiria um treinamento mais abrangente e a captura de uma maior diversidade de padrões e variações presentes nas imagens sísmicas. Isso poderia melhorar ainda mais o desempenho do modelo, levando a resultados mais confiáveis e robustos.

Além das texturas, que foram o foco deste estudo, seria interessante investigar outros tipos de propriedades sísmicas presentes nos bancos de dados utilizados. Por exemplo, explorar características como velocidade, frequência das ondas sísmicas, energia, entre outras, poderia fornecer informações complementares que poderiam melhorar o desempenho da rede neural na segmentação. A análise dessas correlações e atributos adicionais permitiria uma compreensão mais completa dos dados sísmicos e a identificação dos elementos mais relevantes para aprimorar a segmentação e a interpretação geológica.

Em resumo, os trabalhos futuros podem envolver a exploração de novos bancos de dados, o aumento da complexidade do treinamento, o uso de conjuntos de dados maiores e a investigação de outras propriedades sísmicas para aprimorar ainda mais o desempenho do modelo proposto. Essas investigações forneceriam *insights* valiosos para a aplicação de redes neurais em tarefas de segmentação de imagens sísmicas e contribuiriam para avanços significativos na interpretação geológica.

Referências

1 IKELLE, L. T.; AMUNDSEN, L. *Introduction to petroleum seismology*. [S.l.]: Society of Exploration Geophysicists, 2018. Citado 3 vezes nas páginas 5, 17 e 18.

2 MISRA, A. A.; MUKHERJEE, S. *Atlas of structural geological interpretation from seismic images*. [S.l.]: John Wiley & Sons, 2018. Citado 2 vezes nas páginas 5 e 18.

3 GEOPHYSICAL Methods — gsi.ie. <https://www.gsi.ie/en-ie/programmes-and-projects/minerals/activities/mineral-exploration/Pages/Geophysical-Methods.aspx>. [Accessed 05-May-2023]. Citado 2 vezes nas páginas 5 e 19.

4 SABA, J. G. S. Fraturamento hidráulico aplicado a gás de folhelho: aspectos teóricos e práticos. 2014. Citado 2 vezes nas páginas 5 e 19.

5 DI, H.; GAO, D.; ALREGIB, G. Developing a seismic texture analysis neural network for machine-aided seismic pattern recognition and classification. *Geophysical Journal International*, Oxford University Press, v. 218, n. 2, p. 1262–1275, 2019. Citado 2 vezes nas páginas 5 e 20.

6 TRINIDAD, M. J. C. et al. Seismic facies segmentation using atrous convolutional-lstm network. Citado 3 vezes nas páginas 5, 14 e 21.

7 LIMA, G. C. Segmentação de fácies sísmicas com redes neurais. Universidade do Vale do Rio dos Sinos, 2021. Citado 3 vezes nas páginas 5, 18 e 21.

8 GAVOTTI, P. E. *Model-based inversion of broadband seismic data*. Dissertação (Mestrado) — Graduate Studies, 2014. Citado 3 vezes nas páginas 5, 22 e 23.

9 THE velocity spectrum - SEG Wiki — wiki.seg.org. <https://wiki.seg.org/wiki/The_velocity_spectrum>. [Accessed 06-May-2023]. Citado 2 vezes nas páginas 5 e 24.

10 WHAT is a Salt Dome? How do they form? — geology.com. <https://geology.com/stories/13/salt-domes/>. [Accessed 07-Feb-2023]. Citado 2 vezes nas páginas 5 e 25.

11 ATTRIBUTES for fluvial channel interpretation - SEG Wiki — wiki.seg.org. <https://wiki.seg.org/wiki/Attributes_for_fluvial_channel_interpretation>. [Accessed 07-Feb-2023]. Citado 2 vezes nas páginas 5 e 25.

12 LATIFF, A. A.; OSMAN, S.; JAMALUDIN, S. Seismic data analysis to the converted wave acquisition: A case study in offshore malaysia. In: IOP PUBLISHING. *IOP Conference Series: Earth and Environmental Science*. [S.l.], 2016. v. 38, n. 1, p. 012001. Citado 2 vezes nas páginas 5 e 26.

13 SILVA, R. M. et al. Netherlands dataset: A new public dataset for machine learning in seismic interpretation. *arXiv preprint arXiv:1904.00770*, 2019. Citado 4 vezes nas páginas 5, 14, 26 e 27.

14 BARONI, L. et al. Penobscot dataset: Fostering machine learning development for seismic interpretation. *arXiv preprint arXiv:1903.12060*, 2019. Citado 3 vezes nas páginas 5, 28 e 29.

15 LEE, J.-G. et al. Deep learning in medical imaging: general overview. *Korean journal of radiology*, The Korean Society of Radiology, v. 18, n. 4, p. 570–584, 2017. Citado 2 vezes nas páginas 5 e 44.

16 WENG, Y. et al. Nas-unet: Neural architecture search for medical image segmentation. *IEEE Access*, IEEE, v. 7, p. 44247–44257, 2019. Citado 2 vezes nas páginas 6 e 46.

17 LONG, J.; SHELHAMER, E.; DARRELL, T. Fully convolutional networks for semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2015. p. 3431–3440. Citado 5 vezes nas páginas 6, 13, 45, 46 e 47.

18 PEARL, C. *Designing voice user interfaces: principles of conversational experiences*. [S.l.]: "O'Reilly Media, Inc.", 2016. Citado 3 vezes nas páginas 6, 12 e 47.

19 REZENDE, S. O. *Sistemas inteligentes: fundamentos e aplicações*. [S.l.]: Editora Manole Ltda, 2003. Citado 2 vezes nas páginas 11 e 30.

20 LUDERMIR, T. B. Inteligência artificial e aprendizado de máquina: estado atual e tendências. *Estudos Avançados*, SciELO Brasil, v. 35, p. 85–94, 2021. Citado na página 11.

21 PEARL, C. *Designing voice user interfaces: principles of conversational experiences*. [S.l.]: "O'Reilly Media, Inc.", 2016. Citado na página 11.

22 XU, Y. et al. Machine learning in construction: From shallow to deep learning. *Developments in the built environment*, Elsevier, v. 6, p. 100045, 2021. Citado na página 11.

23 READING, H. G. *Sedimentary environments: processes, facies and stratigraphy*. [S.l.]: John Wiley & Sons, 2009. Citado na página 11.

24 SANTOS, M. S. dos; JUNIOR, W. D. de S. O comércio do petróleo: um estudo da estrutura de mercado do setor petrolífero brasileiro no período de 2005 a 2014. *Informe Gepec*, v. 20, n. 1, p. 98–115, 2016. Citado 2 vezes nas páginas 11 e 12.

25 YILMAZ, Ö. *Seismic data analysis: Processing, inversion, and interpretation of seismic data*. [S.l.]: Society of exploration geophysicists, 2001. Citado 5 vezes nas páginas 12, 14, 23, 24 e 25.

26 BARNES, A. E.; LAUGHLIN, K. J. Investigation of methods for unsupervised classification of seismic data. In: *SEG Technical Program Expanded Abstracts 2002*. [S.l.]: Society of Exploration Geophysicists, 2002. p. 2221–2224. Citado na página 12.

27 BARNES, A. E. Redundant and useless seismic attributes. *Geophysics*, Society of Exploration Geophysicists, v. 72, n. 3, p. P33–P38, 2007. Citado na página 12.

28 ONEPETRO. *Seismic facies classification using different deep convolutional neural networks*. Citado na página 12.

29 LI, D. et al. Seismic structure interpretation based on machine learning: A case study in coal mining. *Interpretation*, Society of Exploration Geophysicists and American Association of Petroleum . . . , v. 7, n. 3, p. SE69–SE79, 2019. Citado na página 12.

30 COLÉOU, T.; POUPON, M.; AZBEL, K. Unsupervised seismic facies classification: A review and comparison of techniques and implementation. *The Leading Edge*, Society of Exploration Geophysicists, v. 22, n. 10, p. 942–953, 2003. Citado na página 12.

31 CELECIA, A. et al. Unsupervised machine learning applied to seismic interpretation: Towards an unsupervised automated interpretation tool. *Sensors*, MDPI, v. 21, n. 19, p. 6347, 2021. Citado na página 12.

32 CIVITARESE, D. et al. Semantic segmentation of seismic images. *arXiv preprint arXiv:1905.04307*, 2019. Citado 6 vezes nas páginas 13, 14, 48, 55, 60 e 61.

33 CHEVITARESE, D. S. et al. Deep learning applied to seismic facies classification: A methodology for training. In: EUROPEAN ASSOCIATION OF GEOSCIENTISTS & ENGINEERS. *Saint Petersburg 2018*. [S.l.], 2018. v. 2018, n. 1, p. 1–5. Citado 2 vezes nas páginas 13 e 14.

34 SHALEV-SHWARTZ, S.; BEN-DAVID, S. *Understanding machine learning: From theory to algorithms*. [S.l.]: Cambridge university press, 2014. Citado 2 vezes nas páginas 13 e 17.

35 RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: SPRINGER. *International Conference on Medical image computing and computer-assisted intervention*. [S.l.], 2015. p. 234–241. Citado 3 vezes nas páginas 13, 45 e 46.

36 CHEVITARESE, D. et al. Transfer learning applied to seismic images classification. *AAPG Annual and Exhibition*, 2018. Citado na página 13.

37 CHEVITARESE, D. et al. Seismic facies segmentation using deep learning. *AAPG Annual and Exhibition*, 2018. Citado na página 13.

38 CHEVITARESE, D. S. et al. Efficient classification of seismic textures. In: IEEE. *2018 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2018. p. 1–8. Citado 4 vezes nas páginas 14, 36, 47 e 57.

39 ABID, B.; KHAN, B. M.; MEMON, R. A. Seismic facies segmentation using ensemble of convolutional neural networks. *Wireless Communications and Mobile Computing*, Hindawi, v. 2022, 2022. Citado na página 14.

40 WANG, Z. et al. Improved automating seismic facies analysis using deep dilated attention autoencoders. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. [S.l.: s.n.], 2019. p. 0–0. Citado na página 14.

41 CHEN, X. et al. A stronger baseline for seismic facies classification with less data. *IEEE Transactions on Geoscience and Remote Sensing*, IEEE, v. 60, p. 1–10, 2022. Citado 2 vezes nas páginas 14 e 57.

42 CALHES, D. et al. Simplifying horizon picking using single-class semantic segmentation networks. In: IEEE. *2021 34th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*. [S.l.], 2021. p. 286–292. Citado na página 14.

43 NANDA, N. C. *Seismic data interpretation and evaluation for hydrocarbon exploration and production*. [S.l.]: Springer, 2021. Citado na página 17.

44 ALSADI, H. N. Seismic hydrocarbon exploration. *2D and 3D Techniques, Seismic waves*, Springer, 2017. Citado 2 vezes nas páginas 17 e 19.

45 THOMAS, J. E. *Fundamentos de engenharia de petróleo*. [S.l.]: Interciência, 2001. Citado 2 vezes nas páginas 17 e 19.

46 ONAJITE, E. *Seismic data analysis techniques in hydrocarbon exploration*. [S.l.]: Elsevier, 2013. Citado na página 20.

47 MORAIS, J. M. d. Petróleo em águas profundas: uma história tecnológica da petrobras na exploração e produção offshore. Instituto de Pesquisa Econômica Aplicada (Ipea), 2013. Citado na página 20.

48 BROWN, L. F.; FISHER, W. L. *Seismic stratigraphic interpretation and petroleum exploration*. [S.l.]: AAPG OK, 1980. v. 16. Citado na página 27.

49 PATTERSON, J.; GIBSON, A. *Deep learning: A practitioner's approach*. [S.l.]: "O'Reilly Media, Inc.", 2017. Citado na página 30.

50 BARTHOLMAI, B. J. et al. Pulmonary nodule characterization, including computer analysis and quantitative features. *Journal of thoracic imaging*, LWW, v. 30, n. 2, p. 139–156, 2015. Citado na página 30.

51 LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015. Citado na página 30.

52 GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. <http://www.deeplearningbook.org>. Citado na página 30.

53 KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, v. 25, 2012. Citado na página 30.

54 SOFFER, S. et al. Convolutional neural networks for radiologic images: a radiologist's guide. *Radiology*, Radiological Society of North America, v. 290, n. 3, p. 590–606, 2019. Citado na página 30.

55 FUKUSHIMA, K.; MIYAKE, S.; ITO, T. Neocognitron: A neural network model for a mechanism of visual pattern recognition. *IEEE transactions on systems, man, and cybernetics*, IEEE, n. 5, p. 826–834, 1983. Citado na página 31.

56 LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015. Citado 4 vezes nas páginas 31, 34, 35 e 36.

57 LECUN, Y. et al. Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, v. 2, 1989. Citado na página 31.

58 LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, Ieee, v. 86, n. 11, p. 2278–2324, 1998. Citado na página 31.

59 GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep learning*. [S.l.]: MIT press, 2016. Citado 9 vezes nas páginas 32, 33, 34, 35, 38, 39, 40, 41 e 42.

60 HAYKIN, S.; NETWORK, N. A comprehensive foundation. *Neural networks*, v. 2, n. 2004, p. 41, 2004. Citado na página 36.

61 LI, Y. et al. Seismic data augmentation based on conditional generative adversarial networks. *Sensors*, MDPI, v. 20, n. 23, p. 6850, 2020. Citado na página 41.

62 CRUZ, A. A. da; VELLASCO, M. M. B. R.; PACHECO, M. A. C. Quantum-inspired evolutionary algorithm for numerical optimization. In: IEEE. *2006 IEEE International Conference on Evolutionary Computation*. [S.l.], 2006. p. 2630–2637. Citado na página 42.

63 LUO, P. et al. Towards understanding regularization in batch normalization. *arXiv preprint arXiv:1809.00846*, 2018. Citado na página 42.

64 HE, K. et al. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2016. p. 770–778. Citado 2 vezes nas páginas 42 e 43.

65 COLWELL, R. N. *Manual of photographic interpretation*. [S.l.]: American Society of Photogrammetry, 1960. v. 10. Citado na página 43.

66 SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. Citado na página 43.

67 UHRIG, J. et al. Pixel-level encoding and depth layering for instance-level semantic labeling. In: SPRINGER. *Pattern Recognition: 38th German Conference, GCPR 2016, Hannover, Germany, September 12-15, 2016, Proceedings 38*. [S.l.], 2016. p. 14–25. Citado na página 43.

68 HIJAZI, S. et al. Using convolutional neural networks for image recognition. *Cadence Design Systems Inc.: San Jose, CA, USA*, v. 9, p. 1, 2015. Citado na página 43.

69 HEIJDEN, F. Van der. *Image based measurement systems: object recognition and parameter estimation*. [S.l.]: Wiley, 1994. Citado na página 44.

70 ALBUQUERQUE, V. H. C. d. et al. Sistema de segmentação de imagens para quantificação de microestruturas em metais utilizando redes neurais artificiais. *Matéria (Rio de Janeiro)*, SciELO Brasil, v. 12, p. 394–407, 2007. Citado na página 44.

71 GONZALEZ, R. C. *Richard E. Woods Digital Image Processing, Pearson*. [S.l.]: Prentice Hall, 2018. Citado na página 45.

72 SULTANA, F.; SUFIAN, A.; DUTTA, P. Evolution of image segmentation using deep convolutional neural network: a survey. *Knowledge-Based Systems*, Elsevier, v. 201, p. 106062, 2020. Citado na página 45.

73 SOUSA, I. M. de. Machine learning para segmentação de lesões de esclerose múltipla em imagens de ressonância magnética. 2021. Citado na página 46.

74 MATTOS, A. B. et al. Assessing texture descriptors for seismic image retrieval. In: IEEE. *2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*. [S.l.], 2017. p. 292–299. Citado 2 vezes nas páginas 48 e 55.

75 CHOPRA, S.; ALEXEEV, V. Applications of texture attribute analysis to 3d seismic data. *The Leading Edge*, Society of Exploration Geophysicists, v. 25, n. 8, p. 934–940, 2006. Citado 2 vezes nas páginas 48 e 55.

76 JAIN, A. K. *Fundamentals of digital image processing*. [S.l.]: Prentice-Hall, Inc., 1989. Citado na página 48.

77 GONZALES, R. C.; WINTZ, P. *Digital image processing*. [S.l.]: Addison-Wesley Longman Publishing Co., Inc., 1987. Citado na página 48.

78 KETCHAM, D. J. Real-time image enhancement techniques. In: SPIE. *Image processing*. [S.l.], 1976. v. 74, p. 120–125. Citado na página 48.

79 HUMMEL, R. Image enhancement by histogram transformation. *Unknown*, 1975. Citado na página 48.

80 PIZER, S. An automatic intensity mapping for the display of ct scans and other images. In: *Proceedings of the VIIth International Meeting on Information Processing in Medical Imaging*. [S.l.: s.n.], 1983. p. 276–309. Citado na página 49.

81 PIZER, S. M.; ZIMMERMAN, J. B.; STAAB, E. V. Adaptive grey level assignment in ct scan display. *Journal of Computer Assisted Tomography*, v. 8, n. 2, p. 300–305, 1984. Citado na página 49.

82 PIZER, S. M. et al. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, Elsevier, v. 39, n. 3, p. 355–368, 1987. Citado na página 49.

83 ZUIDERVELD, K. Contrast limited adaptive histogram equalization. *Graphics gems*, Academic Press, p. 474–485, 1994. Citado na página 49.

84 PIZER, S. M. Contrast-limited adaptive histogram equalization: Speed and effectiveness stephen m. pizer, r. eugene johnston, james p. ericksen, bonnie c. yankaskas, keith e. muller medical image display research group. In: *Proceedings of the first conference on visualization in biomedical computing, Atlanta, Georgia*. [S.l.: s.n.], 1990. v. 337, p. 1. Citado na página 49.

85 KURAK, C. Adaptive histogram equalization: a parallel implementation. In: IEEE COMPUTER SOCIETY. *Computer-Based Medical Systems-Proceedings of the Fourth Annual IEEE Symposium*. [S.l.], 1991. p. 192–193. Citado na página 49.

86 RUBINI, C.; PAVITHRA, N. Contrast enhancement of mri images using ahe and clahe techniques. *International Journal of Innovative Technology and Exploring Engineering*, v. 9, n. 2, p. 2442–2445, 2019. Citado na página 50.

87 GONZALES, R. C.; WINTZ, P. *Digital image processing*. [S.l.]: Addison-Wesley Longman Publishing Co., Inc., 1987. Citado na página 51.

88 REZA, A. M. Realization of the contrast limited adaptive histogram equalization (clahe) for real-time image enhancement. *Journal of VLSI signal processing systems for signal, image and video technology*, Springer, v. 38, p. 35–44, 2004. Citado 3 vezes nas páginas 51, 52 e 53.

89 PROJECT F3 Demo 2020, The Netherlands, Offshore, North Sea. <https://terranubis.com/datainfo/F3-Demo-2020>. Accessed: 2022-06-08. Citado na página 57.

90 SILVA, G.; DIAS, J.; FALLETI, L. Análise de arquiteturas de redes neurais para segmentação semântica de dados sísmicos. *SBrT*, 2022. Citado na página 58.