



Pedro Ferreira Alves Pinto

Super Resolução de Imagens Sísmicas

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informático da PUC-Rio.

Orientador: Prof. Sérgio Colcher

Rio de Janeiro
Setembro de 2022



Pedro Ferreira Alves Pinto

Super Resolução de Imagens Sísmicas

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informático da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Prof. Sérgio Colcher

Orientador

Departamento de Informática – PUC-Rio

Prof. Hélio Côrtes Vieira Lopes

Departamento de Informática – PUC-Rio

Prof. Alberto Barbosa Raposo

Departamento de Informática – PUC-Rio

Rio de Janeiro, 30 de Setembro de 2022

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Pedro Ferreira Alves Pinto

Graduado em Ciência da Computação pela Pontifícia Universidade Católica do Rio de Janeiro

Ficha Catalográfica

Ferreira Alves Pinto, Pedro

Super Resolução de Imagens Sísmicas / Pedro Ferreira Alves Pinto; orientador: Sérgio Colcher. – Rio de Janeiro: PUC-Rio, Departamento de Informática, 2022.

v., 51 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática.

Inclui bibliografia

1. Informática – Teses. 2. Super Resolução;. 3. Redes Neurais Profundas;. 4. Processamento de Imagens..
I. Colcher, Sérgio. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

Agradecimentos

Gostaria de agradecer a todos que me ajudaram durante o desenvolvimento deste trabalho.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

Resumo

Ferreira Alves Pinto, Pedro; Colcher, Sérgio. **Super Resolução de Imagens Sísmicas**. Rio de Janeiro, 2022. 51p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

A super resolução (SR) é um tema de suma importância em domínios de conhecimentos variados, como por exemplo a área médica, de monitoramento e de segurança. O uso de redes neurais profundas para a resolução desta tarefa é algo extremamente recente no universo da sísmica, tendo poucas referências, as quais começaram a ser divulgadas há menos de 2 anos. Todavia, a literatura apresenta uma vasta gama de métodos, que utilizam redes neurais para a super resolução de imagens naturais. Tendo isto em vista, o objetivo deste trabalho é explorar tais abordagens aplicadas em dados sísmicos sintéticos de reservatórios. Para isto, foram empregados modelos de importância cronológica na literatura e foram comparados com um método clássico de interpolação e com os modelos da literatura de super resolução de imagens sísmicas. São estes modelos: o SRCNN, o RDN, a abordagem do *Deep Image Prior* e o SAN. Por fim, os resultados apresentam que o PSNR obtido por arquiteturas de projetos no domínio da sísmica equivale a 38.23 e o melhor resultado das arquiteturas propostas 38.62, mostrando o avanço que tais modelos trazem ao campo da sísmica.

Palavras-chave

Super Resolução; Redes Neurais Profundas; Processamento de Imagens.

Abstract

Ferreira Alves Pinto, Pedro; Colcher, Sérgio (Advisor). **Seismic Image Super Resolution**. Rio de Janeiro, 2022. 51p. Dissertação de mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Super resolution (SR) is a topic of notable importance in domains of assorted knowledge, such as the medical, monitoring, and security areas. The use of deep neural networks to solve this task is something extremely recent in the seismic field, with few references, which began to be published less than 2 years ago. However, the literature presents a wide range of methods, using neural networks for the super resolution of natural images. With this in mind, the objective of this work is to explore such approaches applied to synthetic seismic data from reservoirs. For this, models of chronological importance in the literature were used and compared with a classic interpolation method and with models of the literature of super resolution of seismic images. These models are: SRCNN, RDN, the Deep Image Prior approach and SAN. The results show that the PSNR obtained by architectures developed for the seismic domain is equivalent to 38.23 and the best result of the proposed architectures is 38.62, showing the progress that such models bring to the seismic domain.

Keywords

Super-resolution; Deep Learning; Image Processing.

Sumário

1	Introdução	11
1.1	Petrobras	12
1.2	Objetivos do Trabalho	12
1.3	Organização	12
2	Trabalhos Relacionados	13
2.1	M-RUDSR	13
2.1.1	M-RUDSRv2	14
2.2	<i>Deep Learning for Simultaneous Seismic Image Super-Resolution and Denoising</i>	14
3	Fundamentação Teórica	16
3.1	Métricas	16
3.2	<i>Deep Convolutional Network for Image Super-Resolution</i>	17
3.3	<i>Residual Dense Network</i>	19
3.4	<i>Deep Image Prior</i>	21
3.5	<i>Second-Order Attention Network</i>	23
4	Metodologia	28
4.1	Conjunto de Dados	28
4.1.1	Análise exploratória	31
4.1.2	<i>Data augmentation</i>	32
4.1.3	F3	33
4.2	<i>Baseline</i>	33
4.3	<i>Modelos</i>	34
5	Experimentação	37
5.1	Configuração	37
5.1.1	SRCNN	37
5.1.2	RDN	38
5.1.3	DIP	39
5.1.4	SAN	40
5.1.5	M-RUDSRv2	40
5.1.6	Li	41
5.2	Resultados	41
6	Considerações Finais	45
6.1	Trabalhos futuros	46

Lista de figuras

Figura 2.1	Arquitetura M-RUDSR	14
Figura 2.2	Arquitetura Li <i>et al.</i> [43]	15
Figura 3.1	Resultado da comparação entre SRCNN [11] e métodos tradicionais para um exemplo do Set5 [13] com fator de <i>upscaling</i> 3	18
Figura 3.2	Arquitetura da SRCNN [11]	18
Figura 3.3	Arquitetura do RDB [1]	19
Figura 3.4	Arquitetura do RDN [1]	20
Figura 3.5	Imagem retirada de uma publicação do <i>medium</i> escrito por Pratik Katte [36] que ilustra o DIP	23
Figura 3.6	Curva de aprendizado	24
Figura 3.7	Arquitetura da SAN retirada de [9]	25
Figura 4.1	Imagem de reservatório de dimensão 1040 x 7760 gerada artificialmente	28
Figura 4.2	Imagem de reservatório de dimensão 1216 x 6912 gerada artificialmente	29
Figura 4.3	Exemplo de par de imagens em alta e baixa resolução	29
Figura 4.4	Recorte de um contexto na imagem sísmica	30
Figura 4.5	Imagem em baixa resolução, mapa de velocidade e imagem em alta resolução	30
Figura 4.6	Histograma da variabilidade por recorte	31
Figura 4.7	Recorte com desvio padrão 0	32
Figura 4.8	Exemplos de recortes	32
	4.8(a)Recorte com desvio padrão médio	32
	4.8(b)Recorte com desvio padrão máximo	32
Figura 4.9	Representação da localização do F3	33
Figura 4.10	<i>Inline</i> 532	34
Figura 4.11	Exemplo de recorte obtido no F3	35
Figura 4.12	Interpolação bicúbica retirada de [23]	36
Figura 5.1	Comparativo de PSNR por arquitetura da RDN	38
Figura 5.2	Evolução da otimização do DIP	40
Figura 5.3	Instabilidade no treinamento dos modelos ao adicionar dados sísmicos extras	42
	5.3(a)Treinamento sismograma	42
	5.3(b)Treinamento sismograma + velocidade	42
Figura 5.4	Exemplo de saída gerada por cada modelo no <i>dataset</i> Petrobras	43
Figura 5.5	Exemplo de saída gerada por cada modelo no F3	44

Lista de tabelas

Tabela 3.1	Avaliação na escala x2 do PSNR/SSIM	21
Tabela 3.2	Avaliação SAN na escala x2 do PSNR/SSIM	27
Tabela 4.1	Variabilidade por recorte	31
Tabela 5.1	Melhor modelo por arquitetura da RDN	39
Tabela 5.2	PSNR para diferentes arquiteturas adotadas no DIP	39
Tabela 5.3	Avaliação do PSNR dos modelos no conjunto de validação e teste do <i>dataset</i> Petrobras	41
Tabela 5.4	Avaliação do PSNR dos modelos no conjunto de validação e teste do F3	42

Lista de Abreviaturas

SR – Super Resolução

PSNR – Relação Sinal-Ruído de Pico

SSIM – Índice de Similaridade Estrutural

CNN – Redes Neurais Convolutivas

STD – Desvio Padrão

TV – variação tota

MSE – *Mean Squared Error*

MAE – *Mean Absolute Error*

MS-SSIM – *multiscale structural similarity loss*

RDN – *Residual Dense Network*

RDB – *Residual Dense Block*

DIP – *Deep Image Prior*

SAN – *Second-Order Attention Network*

SOCA – *Second-Order Channel Attention*

NLRG – Grupo residual não local

LSRAG – Grupo de atenção residual de origem local

1

Introdução

A tarefa de super resolução [12] consiste em, dada uma matriz de baixa resolução, ampliar suas dimensões de forma artificial com o intuito de obter uma nova matriz de alta resolução. Algoritmos para esta tarefa incluem uma gama de soluções, desde as baseadas em interpolação [10], até métodos baseados em aprendizado [1, 9, 11, 16, 26, 27, 28, 38]. Esta tarefa se mostra relevante em diversos domínios diferentes, como é o caso da área médica [4], bem como da de segurança [5].

Na geofísica, os métodos de super resolução são classificados como métodos baseados em reflexão ou baseados em impedância. Os métodos baseados em reflexão utilizam os mapas de amplitude extraídos do pico de amplitude do sinal sísmico. Por sua vez, os métodos de impedância funcionam em toda a forma de onda.

Devido aos avanços científicos que proporcionam um aumento do poder computacional, e pela resolução de problemas conhecidos, como o *vanishing and exploding gradient*, a utilização de inteligência artificial, em particular as técnicas de aprendizado de máquina, tem ganhado grande ênfase em diversas áreas na academia, principalmente em temas de visão computacional. Dentre o vasto acervo de técnicas utilizadas, as redes neurais profundas, o *Deep Learning*, tem se destacado pelos seus bons resultados [2].

Em 2012, Krizhevsky [3] demonstrou a eficácia do *Deep Learning* ganhando o ImageNet *challenge* [17] com uma margem significativa. Superando assim, resultados que eram considerados estado da arte.

Nas últimas décadas, diversas aplicações para as técnicas de super resolução, como a restauração de imagens e vídeos digitais, foram desenvolvidas. Recentemente, diversos algoritmos para problemas de grande escala e teoremas que definem seus limites foram propostos graças à crescente atenção dada ao tópico.

Assim, considerando o reconhecimento no campo teórico e visando um melhor desempenho na super resolução de dados sísmicos, este trabalho consiste em aliar as técnicas conhecidas de visão computacional aplicadas em imagens naturais à dados sísmicos.

1.1

Petrobras

Este trabalho teve origem como uma tarefa requerida pela Petrobras no projeto "Aprendizado de Máquina na Geofísica" em parceria com o Departamento de Informática da PUC-Rio, com o intuito de desenvolver pesquisas e ferramentas de aprendizado de máquina e aprendizado profundo na área de geofísica, especificamente voltadas para a integração de rocha, perfil e sísmica. Esta parceria se mostrou bastante produtiva gerando contribuições para a comunidade científica como em [18, 19, 20, 21, 22]. Os principais conjuntos de dados utilizados foram produzidos e disponibilizados pela Petrobras para pesquisas no contexto do projeto.

1.2

Objetivos do Trabalho

O objetivo deste trabalho é aplicar um grupo de algoritmos relevantes na literatura de super resolução de imagens em cima de dados sísmicos, com o intuito de mostrar a relevância e melhoria que estes podem representar dentro do domínio da sísmica.

Neste trabalho, a tarefa de super resolução de dados sísmicos consiste em aumentar artificialmente a taxa de amostragem da impedância do sinal sísmico a partir de uma amostra que possui uma taxa de amostragem menor.

Em visão computacional, os algoritmos de super resolução geralmente ampliam a resolução das imagens em 2 ou 4 vezes a sua resolução. Devido a isto, foi definido, para um primeiro momento, a pesquisa e desenvolvimento de soluções focadas em aumentar a resolução de imagens sísmicas migradas em 2 vezes.

1.3

Organização

Feita a introdução do projeto, esta seção é responsável por apresentar o conteúdo dos demais capítulos. Assim, o capítulo 2 apresenta os trabalhos relacionados ao tema de super resolução de dados sísmicos. Em seguida, o capítulo 3 busca construir a base teórica necessária para uma maior compreensão do projeto, passando por temas como métricas utilizadas e os modelos relevantes da literatura. O capítulo 4, por sua vez, apresenta os *datasets* utilizados, bem como a descrição dos modelos adotados durante a pesquisa. O capítulo 5 descreve a experimentação e os resultados do projeto. O capítulo 6 traz as conclusões. E, por último, temos a bibliografia.

2

Trabalhos Relacionados

Uma busca na literatura relacionada ao uso de redes neurais profundas para super resolução de dados sísmicos, mostra que o seu uso é extremamente recente, só iniciando nos últimos 2 anos e tendo a maioria dos trabalhos publicados no ano de 2021.

Uma dificuldade recorrente apresentada nos trabalhos é a falta de dados públicos para treinamento e avaliação dos modelos. Devido a isto, os trabalhos avaliados utilizaram um conjunto próprio de dados sintéticos providos por organizações privadas. Isto faz com que seja inviável a reprodução da experimentação e treinamento dos modelos propostos.

2.1 M-RUDSR

Em 2020, foi publicado um artigo chamado *Multitask Learning for Super-Resolution of Seismic Velocity Model* [42], onde foi proposto um modelo de redes neurais profundas, denominado M-RUDSR, para a resolução de tarefas de ampliação da resolução e para produzir a imagem de arestas de modelos de velocidade sísmicos. Ademais, foram propostos outros dois modelos para servirem como comparativo, o RUDSR e o M-RUSR.

O M-RUDSR consiste no uso de *skip connection* residuais globais, uma estrutura *encoder-decoder* baseada em U-Net [35] e blocos densos de *skip connection*. Todas as camadas convolutivas fora dos blocos densos apresentam *kernel* de tamanho 3x3. Os blocos densos são compostos por uma camada convolutiva 1x1, com uma função de ativação ReLu, seguida por uma nova camada convolutiva 3x3. A Figura 2.1 apresenta a arquitetura do sistema.

Por sua vez, o RUDSR é uma versão do M-RUDSR para realizar apenas a tarefa de SR. O M-RUSR corresponde a uma versão simplificada do M-RUDSR sem a utilização do bloco denso de *skip connection*.

A função de *loss* consiste no uso do MSE, também conhecido como *loss* L_2 , e no uso da variação total (TV).

Os modelos foram treinados em um conjunto sintético de dados gerados pelo próprio grupo de pesquisa, o *Easy-Signal Laboratory*. A avaliação foi

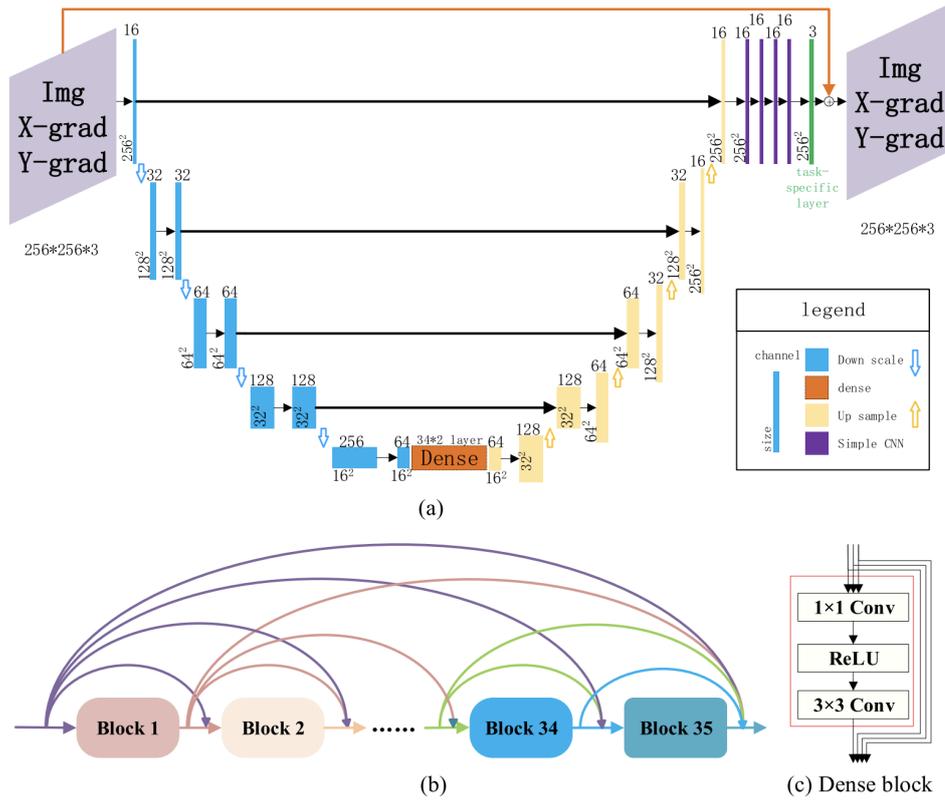


Figura 2.1: Arquitetura M-RUDSR

executada em dados sintéticos gerados através do Marmousi [25] e em dados reais da bacia de Sichuan na China.

2.1.1 M-RUDSRv2

Um ano após esta publicação, o mesmo grupo de pesquisadores publicou um novo artigo denominado *Super-Resolution of Seismic Velocity Model Guided by Seismic Data* [44] onde foi apresentado o M-RUDSRv2.

O objetivo do M-RUDSRv2 é a utilização de dados sísmicos para melhorar os resultados da super resolução dos modelos de velocidade. Para isto, foi proposta uma abordagem onde os dados sísmicos são combinados com os modelos de velocidade com o intuito trazer mais informação para a rede a ser treinada.

2.2 Deep Learning for Simultaneous Seismic Image Super-Resolution and Denoising

Também em 2021, foi proposto um modelo de CNN para a realização simultânea da tarefa de *denoising* e a tarefa de super resolução de dados

sísmicos Li *et al.* [43].

A arquitetura proposta se assemelha ao M-RUDSR, ambas equivalem a uma variação da U-net. Entretanto, no caso da rede proposta por Li, a entrada corresponde à imagem em baixa resolução e uma camada subpixel [41] é introduzida para efetuar a etapa de *up-sampling*. Além disso, esta arquitetura apresenta blocos residuais após a camada de subpixel, onde cada bloco é composto por duas camadas convolutivas seguidas por uma de *batch normalization* e uma ReLu. A rede descrita esta ilustrada na Figura 2.2.

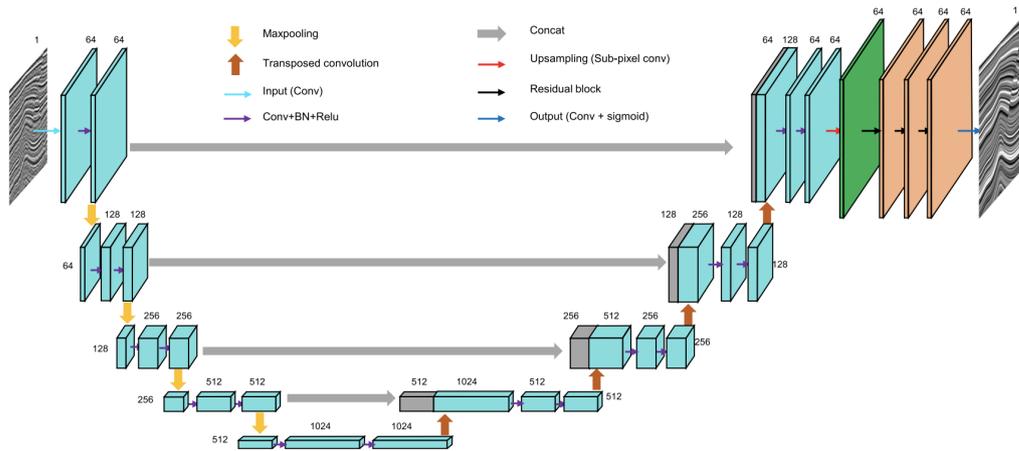


Figura 2.2: Arquitetura Li *et al.* [43]

Em relação a função de *loss*, foi feita uma agregação do erro absoluto médio, também conhecido como L_1 , e o *multiscale structural similarity loss* (MS-SSIM) [7].

O treinamento do modelo foi realizado em um grupo de dados sintéticos gerados pelo grupo de pesquisadores. Sua avaliação, no entanto, foi feita em dados reais.

3 Fundamentação Teórica

Este capítulo busca construir a base teórica necessária para uma maior compreensão do projeto. Para isso, foi explorado bibliografias que ilustram a evolução tecnológica de Super Resolução. Foram utilizados 4 modelos que adotam técnicas que se mostraram notáveis nos avanços das pesquisas sobre redes neurais ao tempo de suas publicações.

Devido a alta visibilidade desta tarefa, trabalhos são publicados constantemente fazendo com que o estado da arte da Super Resolução de imagens naturais mude constantemente. O artigo *Second-Order Attention Network* [9] apresentava os resultados estado da arte para um conjunto de *benchmarks* durante o início da pesquisa deste trabalho. Com isto, foi o último modelo a ser adotado.

Inicialmente, será feita uma breve explicação sobre as métricas de avaliação dos modelos. Seguida de uma explicação das técnicas de redes neurais selecionadas da literatura.

3.1 Métricas

As métricas tradicionais para avaliação de qualidade de modelos de super resolução são a Relação Sinal-Ruído de Pico (PSNR) e o Índice de Similaridade Estrutural (SSIM) [6].

A PSNR consiste na relação entre a máxima energia de um sinal e o ruído que afeta sua representação original. Desta forma, quanto maior o PSNR, menor o erro. Assim, em virtude de muitos sinais terem amplitude dinâmica, a PSNR é normalmente expressa por uma escala logarítmica em escala decibel, conforme a seguinte equação:

$$PSNR(f, g) = 10 \log_{10}(MAX_i^2 / MSE(f, g)) \quad (3-1)$$

onde,

$$MSE(f, g) = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (f_{ij} - g_{ij})^2 \quad (3-2)$$

Sendo MAX_i o maior valor possível. Uma vez que estamos tratando de imagens, este valor corresponde a 255.

Por sua vez, o SSIM, ao invés de utilizar métodos tradicionais de somatórios, consiste em uma métrica de medida de similaridade entre duas imagens, levando em consideração elementos como a distorção da luminância, a distorção do contraste e a estrutura da imagem, conforme a fórmula a seguir:

$$SSIM(f, g) = l(f, g)c(f, g)s(f, g) \quad (3-3)$$

onde,

$$\begin{aligned} l(f, g) &= \frac{2\mu_f\mu_g + C_1}{\mu_f^2 + \mu_g^2 + C_1} \\ c(f, g) &= \frac{2\sigma_f\sigma_g + C_2}{\sigma_f^2 + \sigma_g^2 + C_2} \\ s(f, g) &= \frac{\sigma_{fg} + C_3}{\sigma_f\sigma_g + C_3} \end{aligned} \quad (3-4)$$

Neste projeto, não adotaremos a métrica SSIM pois o seu cálculo utiliza propriedades intrínsecas de imagens naturais.

3.2

Deep Convolutional Network for Image Super-Resolution

Em 2014, Dong [11], pela primeira vez abordou redes neurais convolucionais (CNN) [14] para a tarefa de Super Resolução de imagens. Os seus resultados mostraram uma melhora significativa em relação a métodos tradicionais como apresentado na Figura 3.1.

Sua abordagem aprende um mapeamento *end-to-end* entre imagens de baixa e alta resolução. Este mapeamento é representado através de uma rede neural convolucional, denominada SRCNN, que tem como entrada a imagem de baixa resolução e como saída a de alta.

O algoritmo da SRCNN consiste nas seguintes etapas: Primeiramente, a imagem de baixa resolução passa por uma etapa de pré processamento, onde é aplicada uma interpolação bicúbica para deixar a imagem no tamanho desejado.

Feito isto, esta imagem passa por uma camada de CNN, onde serão extraídos atributos da imagem. A camada seguinte é responsável por mapear estes atributos de maneira não linear à representação em alta resolução. Por último, temos uma terceira camada para combinar as predições dentro de uma

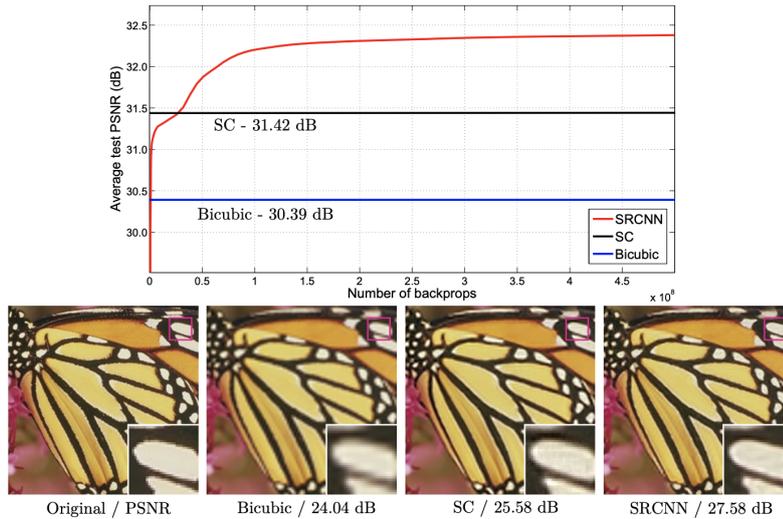


Figura 3.1: Resultado da comparação entre SRCNN [11] e métodos tradicionais para um exemplo do Set5 [13] com fator de *upscaling* 3

vizinhança espacial para gerar a imagem final em alta resolução. A Figura 3.2, apresenta de maneira gráfica as etapas apresentadas.

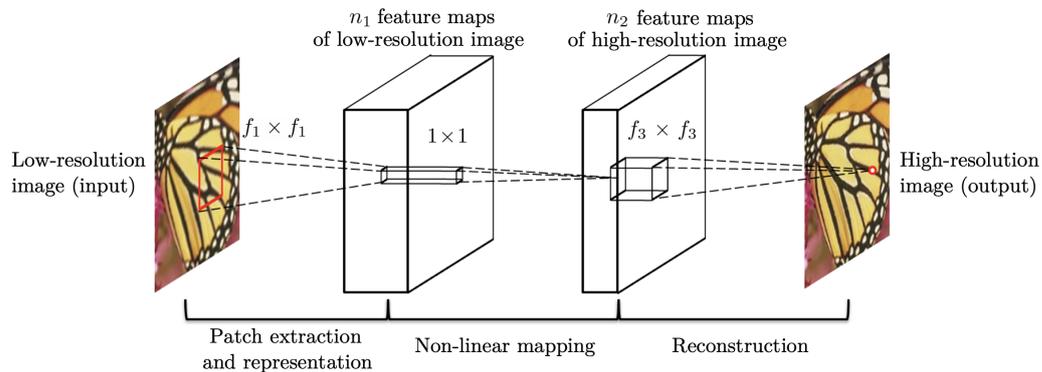


Figura 3.2: Arquitetura da SRCNN [11]

A configuração padrão utilizada é formada por funções de ativação ReLU, CNN com *kernel* de tamanho 9x9 na primeira camada, com um total de 64 canais, tamanho 1x1 na segunda camada, com um total de 32 canais e tamanho 5x5 na última camada.

Em relação a função de *loss*, é utilizado o *Mean Squared Error* (MSE) entre a imagem original e a imagem produzida pela rede, devido ao fato do MSE favorecer altos valores de PSNR.

A avaliação dos resultados do SRCNN foi feita em 2 *datasets* considerados *benchmarks*: o Set5 [13] e o Set14 [15]. Em ambos os conjuntos de dados, o modelo proposto por Dong apresentou os melhores resultados até a data de

publicação do artigo. Obtendo assim, PSNR médio equivalente a 30.09 e 29.00 respectivamente.

3.3

Residual Dense Network

Em 2018, Zhang [1] apresentou em seu artigo um modelo com resultados que superavam o estado da arte para a tarefa de SR. Inicialmente, foi reconhecido o grande sucesso das CNNs na super resolução de imagens. Entretanto, Zhang alegou que a maioria dos modelos profundos de CNN não estavam fazendo pleno uso das *features* hierárquicas da imagem em baixa resolução. Deste modo, não estavam atingindo a potencial performance oferecida pelos modelos.

Tendo isto em vista, foi proposta uma rede residual densa, conhecida em inglês como *residual dense network*, RDN. Com base no conceito de blocos residuais, proposto na [24], o RDN busca construir *residual dense block* (RDB) para extrair um vasto número de *features* locais através de densas camadas convolutivas. A fusão destes atributos locais permite o aprendizado de novas *features* relevantes assim como a estabilização do treinamento do modelo. A Figura 3.3 ilustra os conceitos apresentados.

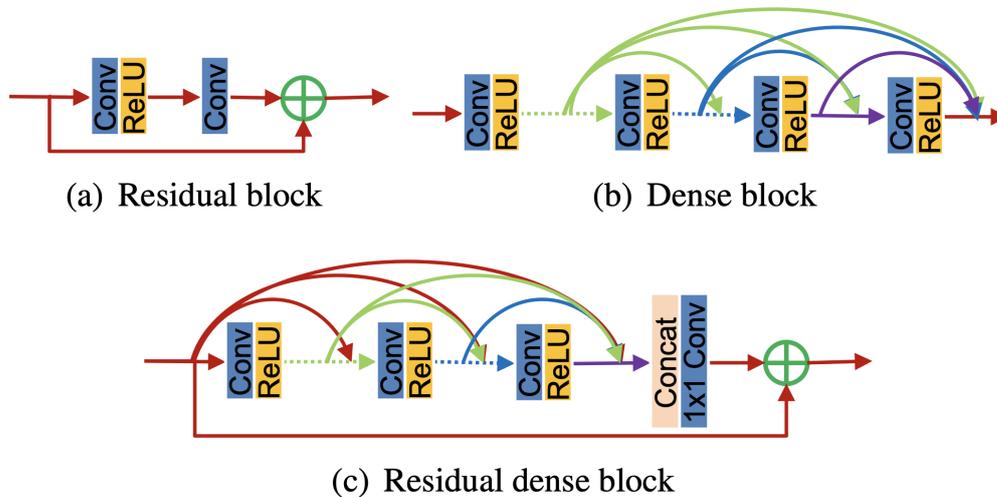


Figura 3.3: Arquitetura do RDB [1]

Após a etapa de obtenção das *features* locais, é feita uma fusão global a partir das *features* obtidas em cada bloco RDB para aprender atributos hierárquicos globais da imagem.

O conceito de blocos residuais já havia sido explorado anteriormente por outros trabalhos, como é o caso da SRResNet [26], de Lim com sua rede EDSR[27] e da MemNet, proposto por Tai [28]. Contudo, as camadas convolutivas locais não tem acesso direto às camadas seguintes. Com isso, eles não fazem pleno uso das informações de todas as camadas dentro do modelo.

A arquitetura da RDN, como apresentado na Figura 3.4, constitui-se em quatro partes:

1. Rede para extração de *features* rasas (SFENet)
2. Blocos RDB
3. Fusão das *features* densas (DFF)
4. Rede de up-sampling (UPNet)

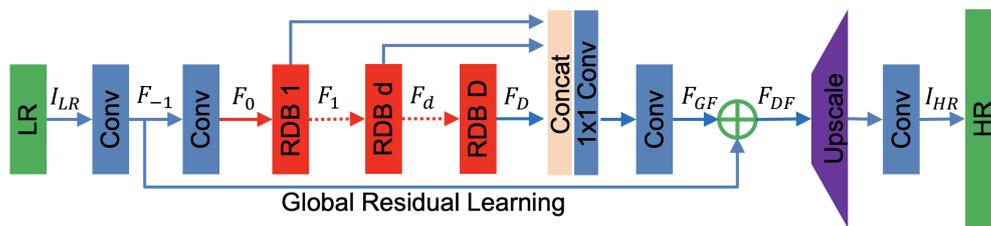


Figura 3.4: Arquitetura do RDN [1]

A etapa referente a SFENet equivale a duas camadas convolutivas para extração das *features* rasas da imagem de baixa resolução. A saída da primeira camada é usada como entrada da segunda camada convolutiva e, após, na fusão global de *features*. Por sua vez, a segunda camada é usada como entrada para os blocos RDB.

O bloco RDB, Figura 3.3, possui um mecanismo de memória contígua, onde os atributos de saída de uma camada são passados para as demais camadas subsequentes, preservando a natureza *feed-forward* das redes neurais e extraindo *features* locais densas.

A próxima etapa da extração dos atributos da RDB é a fusão das *features* locais. Nesta etapa, todas as *features* produzidas dentro do bloco RDB são concatenadas à entrada e, então, passam por uma camada convolutiva 1x1 para filtrar a informação essencial.

Para a conclusão do RDB, é feito um aprendizado residual local, em que a entrada do bloco é somada à saída da fusão das *features*.

A terceira parte da arquitetura da RDN, o DFF, é aplicada para explorar de maneira global as *features* hierárquicas. Esta etapa consiste em uma fusão global das *features* e um aprendizado residual global.

A fusão global das *features* é aplicada de maneira semelhante à fusão local do RDB. Trata-se da concatenação da saída de todos os RDBs seguida pela aplicação da camada convolutiva 1x1. Ao final, uma nova camada 3x3 é introduzida para extrair as *features* para a última etapa.

Por sua vez, o aprendizado residual global é a soma da saída da fusão global das *features* e os atributos da primeira camada da extração de *features* rasas.

Finalmente, o UPNet se resume a adoção do ESPCN [29] seguido por uma camada de rede convolutiva.

As configurações adotadas pela implementação são 64 canais para as camadas de fusão de *features* local e global e 20 blocos RDB contendo 6 camadas convolutivas com 32 canais cada. Todas as camadas relacionadas a extração de *features* utilizam como função de ativação a ReLU. A camada final da rede possui 3 canais, visto que a saída da rede é uma imagem RGB.

Em relação a função de *loss*, foi escolhido o uso da L_1 , também conhecida como erro absoluto médio ou MAE. Sua escolha vou embasada no artigo [27], que mostra que a L_1 possui uma melhor performance e convergência do que a L_2 , o MSE. Ademais, é empregado o otimizador Adam [30] e uma taxa de aprendizado inicial de 0.0001 que é dividida pela metade a cada 200 épocas.

Além de utilizar os *datasets* Set5 [13], Set14 [15], adotados pela SRCNN, o RDN também é avaliado nos conjuntos de dados BSD100 [31], Urban100 [32] e Manga109 [33].

Os resultados obtidos no aumento de resolução com escala x2, superam a interpolação bicúbica, o SRCNN e o estado da arte anterior, MDSR [27], como apresentado na Tabela 3.1.

<i>Dataset</i>	Interpolação	SRCNN	MDSR	RDN
Set 5	33.66/0.9299	36.66/0.9542	38.11/0.9602	38.24/0.9614
Set 14	30.24/0.8688	32.45/0.9067	33.85/0.9198	34.01/0.9212
BSD100	29.56/0.8431	31.36/0.8879	32.29/0.9007	32.34/0.9017
Urban100	26.88/0.8403	29.50/0.8946	32.84/0.9347	32.89/0.9353
Manga109	30.80/0.9339	35.60/0.9663	38.96/0.9769	39.18/0.9780

Tabela 3.1: Avaliação na escala x2 do PSNR/SSIM

3.4

Deep Image Prior

Poucos meses depois, ainda em 2018, Ulyanov publicou o artigo *Deep Image Prior* (DIP) [8]. A sua publicação propõem um método para a resolução de três tarefas de reconstrução de imagem: a remoção de ruídos, o *inpainting* e a super resolução.

Parte do interesse da comunidade científica em relação a este artigo foi o uso contrastante de CNNs aos demais usos apresentadas por outros pesquisadores. O modelo de redes convolutivas utilizado não é treinado.

Normalmente a alta performance das CNNs é atribuído a sua habilidade de capturar informações prévias de imagens através de um grande conjunto de exemplos. Entretanto, Ulyanov argumenta que, ao contrário do que é acreditado, a estrutura de uma rede geradora é suficiente para capturar uma vasta quantidade de estatísticas da imagem e identificar a informação perdida no processo de degradação da imagem.

O DIP trata o problema da super resolução como um caso onde a distribuição $f\theta(z)$, aprendida por uma rede geradora, foi condicionada em uma observação corrompida x_0 , gerando uma imagem degradada.

Para o entendimento do método é antes necessário observar o problema de recuperação de imagens como um problema de minimização de energia, seguindo a seguinte formula:

$$x^* = \operatorname{argmin}_x E(x; x_0) + R(x) \quad (3-5)$$

Sendo x^* a imagem restaurada; x_0 imagem degradada; x a imagem limpa; E no artigo é usado como a entropia cruzada e $R(x)$ é um regularizador, que neste artigo é tratado como 0.

O objetivo do método proposto é investigar o a priori implicitamente capturado por uma rede geradora, antes da etapa de aprendizado do modelo. Para isso, a rede neural é interpretada como uma parametrização da equação $x = f\theta(z)$. Onde z é um tensor de tamanho 32 x altura da imagem x largura da imagem preenchido com ruído uniforme e θ os parâmetros da rede geradora. Aplicando esta ideia à formula 3-5 temos:

$$\theta^* = \operatorname{argmin}_\theta E(f\theta(z); x_0) \quad (3-6)$$

A minimização de θ^* é feita por meio do uso do gradiente descendente, começando de uma inicialização randômica dos parâmetros. Feita a minimização, a imagem restaurada, é obtida através de $x^* = f\theta^*(z)$.

Através de experimentos conduzidos no artigo, os autores mostram que o uso do gradiente descendente para a otimização do problema faz com que a rede resista a soluções ruidosas e desça muito mais rapidamente para imagens de aparência natural. A Figura 3.6 apresenta um gráfico que ilustra o que foi dito.

Considerando a necessidade da imagem degradada x_0 ter as mesmas dimensões que x , é aplicada uma interpolação, no caso Lanczos, para garantir o mesmo tamanho. Em outras palavras, a degradação da imagem neste caso

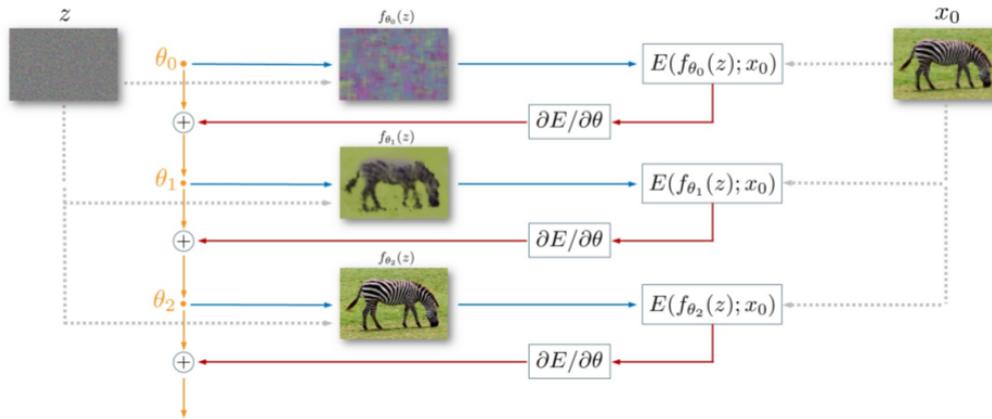


Figura 3.5: Imagem retirada de uma publicação do *medium* escrito por Pratik Katte [36] que ilustra o DIP

pode ser vista como a diferença entre a interpolação e a imagem original.

A avaliação dos modelos decorreu da aplicação do método no *datasets* de *benchmark* Set5 [13] e Set14 [15] em um aumento de resolução de 4x. Durante as experimentações, a rede utilizada foi uma U-net [35] com *skip-connections*.

Os resultados obtidos foram 29.90 e 27.00 de PSNR no Set5 e Set14, sendo os da interpolação bicúbica 28.43 e 26.05, e os do estado da arte durante a elaboração da pesquisa 32.10 e 28.53.

O fato dos resultados ficarem superiores à interpolação bicúbica mostra a eficácia do método. Por outro lado, existe uma vão entre os resultados considerados estado da arte e os do DIP.

3.5

Second-Order Attention Network

Em meados de 2019, foi publicado por um grupo de pesquisadores em Hong Kong, o artigo *Second-Order Attention Network* (SAN) [9], em que mecanismos de atenção foram incorporados, trazendo uma melhoria no uso e no aprendizado da correlação das *features*.

Os autores argumentam que a maioria dos novos métodos propostos focam em projetar arquiteturas mais amplas ou profundas, porém, não exploram as correlações das *features* de camadas intermediárias. Com isto, não utilizam a íntegra do poder de representação das CNNs.

Com o intuito de alcançar maior aproveitamento das *features* aprendidas pelo modelo, foi desenvolvido um bloco de atenção de canal de segunda ordem, em tradução livre de *second-order channel attention* (SOCA), para, através de estatísticas de segunda ordem de *features*, redimensionar de forma adaptativa as *features* de canais. Ademais, foi criada também uma estrutura de grupo residual não local (NLRG) que, além da utilização de operações não locais para

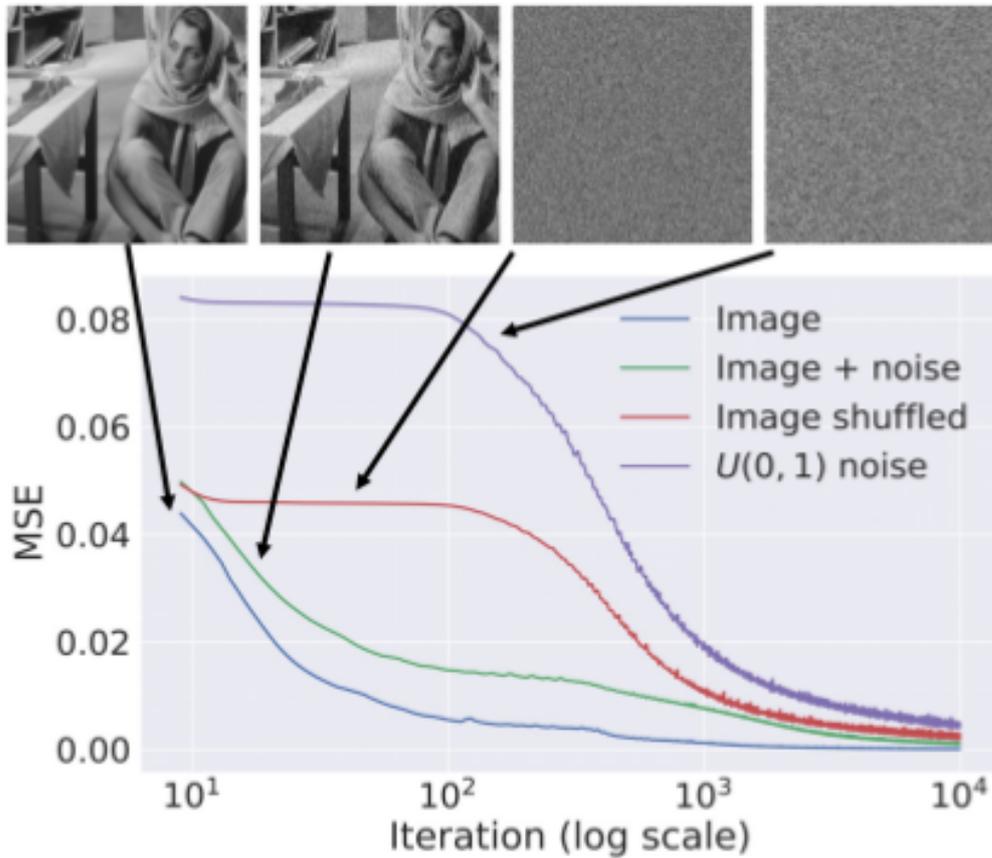


Figura 3.6: Curva de aprendizado

captura de informações espaciais distantes, possui grupos de atenção residual de origem local (LSRAG) para um aumento da representação abstrata das *features*.

Os mecanismos de atenção já haviam sido anteriormente adotados para aprimorar as arquiteturas e, conseqüentemente, os resultados de CNNs na super resolução de imagens. A rede RCAN [38] alia CNNs profundas ao mecanismo de atenção *squeeze and excitation* [37] para melhorar a performance da SR. Entretanto, o mecanismo de atenção usado manipula apenas estatísticas de primeira ordem, como o *global average pooling*, e, como dito anteriormente, não utilizam a íntegra do poder de representação das CNNs.

Outro ponto apontado no artigo é o fato de *features* de maior abstração de informação serem mais informativas para a reconstrução de imagens em alta resolução.

A arquitetura do SAN, como ilustrado na Figura 3.7, é formada por quatro partes:

1. Extração de *features* rasas
2. Extração de *features* baseado em NLRG

3. Módulo de *upscale*
4. Reconstrução

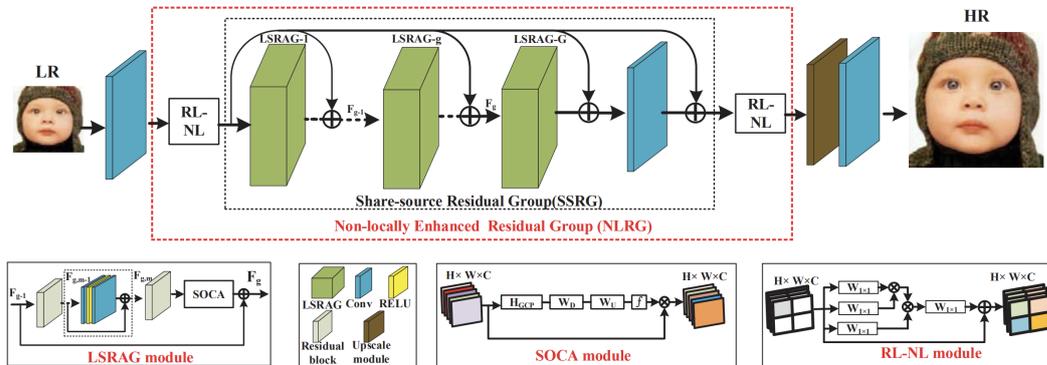


Figura 3.7: Arquitetura da SAN retirada de [9]

Inspirado pelo trabalho da RDN, a etapa de extração de *features* rasas consiste em apenas uma camada convolutiva.

Em seguida, estas *features* rasas servem de entrada à um bloco de extração de *features* latentes, o NLRG. Este, por sua vez, é composto por diversos módulos *non-local* [39] em nível de região (RL-NL) e por um grupo residual *share-source* (SSRG), contendo G módulos LSRAG com *skip connections* entre eles. Deste modo, uma rede de grande profundidade é construída, capacitando o aprendizado de *features* de maior abstração.

No RL-NL, para contornar algumas limitações das redes neurais *non-local* [39], a entrada é dividida em um *grid* de regiões com k^2 blocos de mesmo tamanho, onde cada bloco será processado pelas camadas subsequentes. Desta forma, é possível explorar a correlação espacial entre as *features* e capturar dependências de longo alcance.

Os blocos LSRAG contêm M blocos residuais simplificados com *skip connections* locais, seguidos do módulo de atenção SOCA para explorar a interdependência das *features*.

Inspirados por trabalhos como [40], que mostram que a utilização de estatísticas de secunda ordem auxiliam mais na representação discriminativa da informação, os autores propuseram um bloco SOCA.

Inicialmente, o SOCA, seguindo os trabalhos que motivaram sua criação, começa efetuando um normalização de covariância \hat{Y} , que é dada pela potência dos autovalores obtidos pela decomposição de autovalores da matriz de covariância Σ . A equação a seguir apresenta as etapas descritas.

$$\Sigma^\alpha = X\bar{I}X^T \quad (3-7)$$

onde, X é uma matriz com WH *features* de C dimensões, sendo H e W a altura e largura da entrada, e $\bar{I} = \frac{1}{WH}(I - \frac{1}{WH}1)$ sendo I a matriz identidade e 1 uma matriz preenchida de valores 1.

$$\Sigma = U\Lambda U^T \quad (3-8)$$

$$\hat{Y} = \Sigma^\alpha = U\Lambda^\alpha U^T \quad (3-9)$$

Para que a normalização ocorra, α , por sua vez, é o valor $1/2$. Sendo assim, a normalização diminui não linearmente os autovalores maiores que 1 e aumenta os menores que 1.

A matriz de covariância normalizada é utilizada como descritor dos canais através de um *global covariance pooling* (GCP). Este GCP, diferente do *global average pooling*, captura estatísticas de maiores ordens.

Além disso, o SOCA aplica duas camadas convolutivas e uma função *sigmoid* para servir como *gate*. O resultado desta operação é, por sua vez, multiplicado à entrada do bloco SOCA, servindo assim como um mecanismo de atenção.

A etapa de *upscale* segue o que foi feito na UPNet do RDN e adota o ESPCN [29].

Por último, as *features* redimensionadas são mapeadas à imagem em alta resolução através de uma camada de rede convolutiva com três canais correspondentes ao RGB da imagem.

As configurações da rede consistem em 20 blocos LSRAG com 10 blocos residuais cada, um módulo RL-NL na entrada e outro na saída da SSRG. As CNN no SOCA possuem tamanho 1×1 e reduzem em 16 vezes os canais de entrada. As demais camadas convolutivas possuem tamanho 3×3 e 64 canais.

Os experimentos foram feitos nos conjuntos de dados Set5, Set14, BSD100, Urban100 e Manga109. Os resultados obtidos se mostram competitivos e, em alguns casos, superiores aos demais modelos da literatura. Um comparativo é apresentado na Tabela 3.2. Foi adicionado o modelo RCAN [38] por possuir resultados equivalentes e adotar também mecanismos de atenção.

<i>Dataset</i>	Interpolação	SRCNN	RDN	RCAN	SAN
Set 5	33.66/0.9299	36.66/0.9542	38.24/0.9614	38.27/0.9614	38.31/0.9620
Set 14	30.24/0.8688	32.45/0.9067	34.01/0.9212	34.11/0.9216	34.07/0.9213
B100	29.56/0.8431	31.36/0.8879	32.34/0.9017	32.41/0.9026	32.42/0.9028
Urban100	26.88/0.8403	29.50/0.8946	32.89/0.9353	33.34/0.9384	33.10/0.9370
Manga109	30.80/0.9339	35.60/0.9663	39.18/0.9780	39.43/0.9786	39.32/0.9792

Tabela 3.2: Avaliação SAN na escala x2 do PSNR/SSIM

4 Metodologia

Este capítulo descreve o processo de obtenção do *dataset* utilizado no projeto, assim como os processamentos feitos nos dados e uma breve análise exploratória. Ademais, as seções seguintes apresentam especificações referentes ao *baseline* e aos modelos utilizados pelo sistema.

4.1 Conjunto de Dados

Este *dataset* foi construído com base em duas imagens completas de reservatórios produzidas artificialmente e disponibilizadas pela Petrobras com o intuito do desenvolvimento de pesquisas no contexto do projeto "Aprendizado de Máquina na Geofísica".

Estas imagens possuem tamanhos $n_1 \times n_2$, sendo n_1 o número de amostras (linhas da imagem) e n_2 o número de traços (colunas da imagem). Cada imagem possui intervalos próprios de amostragem nas direções X e Y que são denotados por d_1 e d_2 . Além disso, foram disponibilizados os mapas de velocidade de cada um dos reservatórios.

O primeiro reservatório apresenta dimensão 1040 x 7760 e seu valor de amostragem corresponde a $d_1 = 5$ e $d_2 = 5$. A Figura 4.1 ilustra esta imagem.

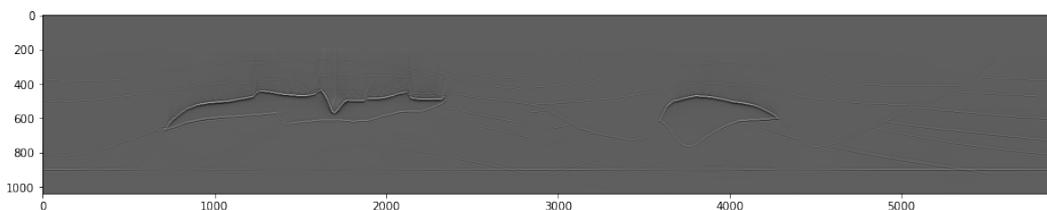


Figura 4.1: Imagem de reservatório de dimensão 1040 x 7760 gerada artificialmente

Por sua vez, o segundo reservatório apresenta n_1 e n_2 de 1216 x 6912 respectivamente. Seus valores de amostragem são $d_1 = 5$ e $d_2 = 5$. A Figura 4.2 apresenta uma visualização desta imagem.

O *dataset* para a tarefa de super resolução consiste em pares de imagens sísmicas migradas em baixa resolução e suas correspondentes imagens na resolução almejada.

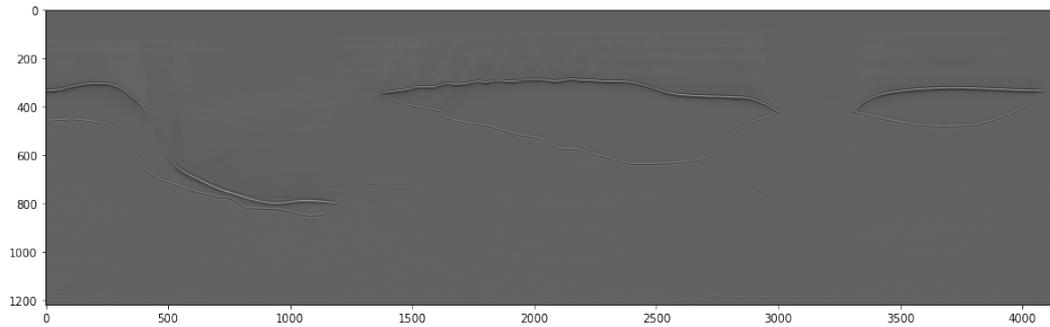


Figura 4.2: Imagem de reservatório de dimensão 1216 x 6912 gerada artificialmente

A Figura 4.3 apresenta um exemplo de par de imagens presente no conjunto de dados, onde na esquerda temos um exemplo em resolução 64 x 64 e na direita uma amostra com a qualidade degradada em 2 vezes, obtendo um tamanho de 32 x 32.

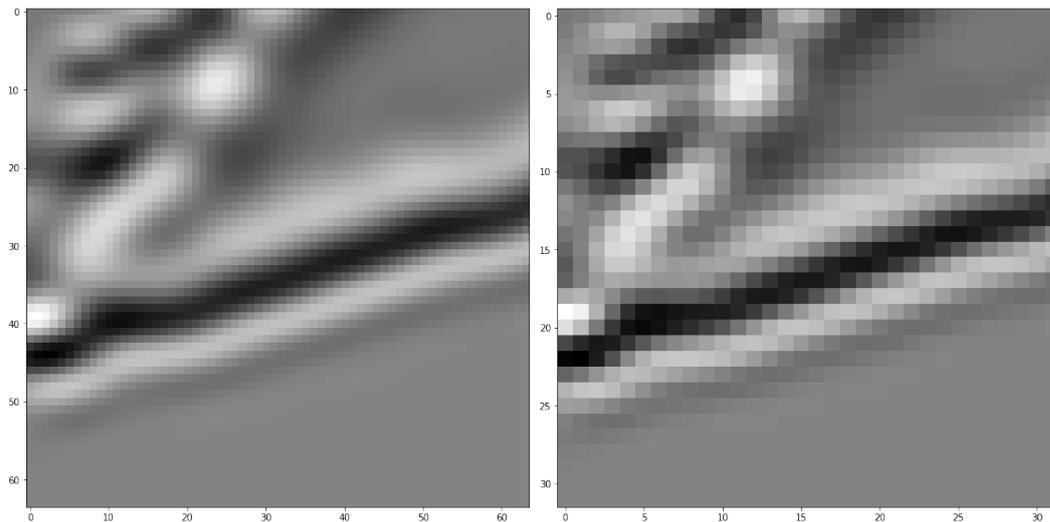


Figura 4.3: Exemplo de par de imagens em alta e baixa resolução

Assim, utilizando o método, geralmente adotado para geração de conjunto de dados para super resolução em visão computacional, construímos o nosso conjunto de dados da seguinte maneira: Primeiramente, os modelos de sal são fragmentados em uma grade de contextos de tamanho 64 x 64. Em seguida, reduzimos a resolução destes contextos pela metade por um algoritmo de interpolação pela média local. Por fim, os contextos em alta resolução são utilizados como meta para a saída do modelo e os contextos degradados são utilizados como entrada. Na Figura 4.4, ilustramos o recorte de um contexto na imagem sísmica.

Como foi apresentado no artigo do M-RUDSRv2[44], o uso de informações adicionais trazem melhorias nas métricas de avaliação. Uma vez que foi

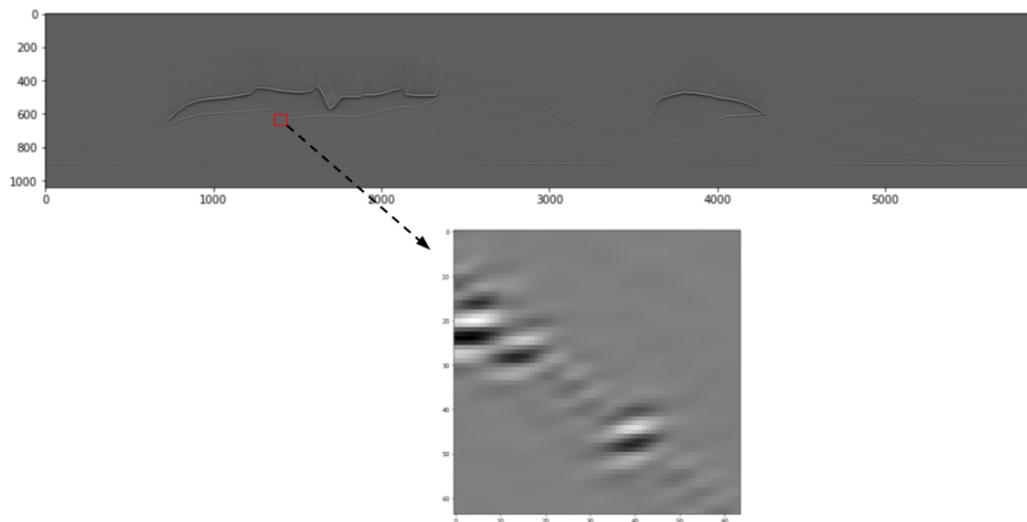


Figura 4.4: Recorte de um contexto na imagem sísmica

disponibilizado o mapa de velocidade, ele será fragmentado conforme o método apresentado e será utilizado como entrada para os modelos. A Figura 4.5 apresenta a imagem em baixa resolução, mapa de velocidade e a imagem em alta resolução.

PUC-Rio - Certificação Digital N° 1921386/CA

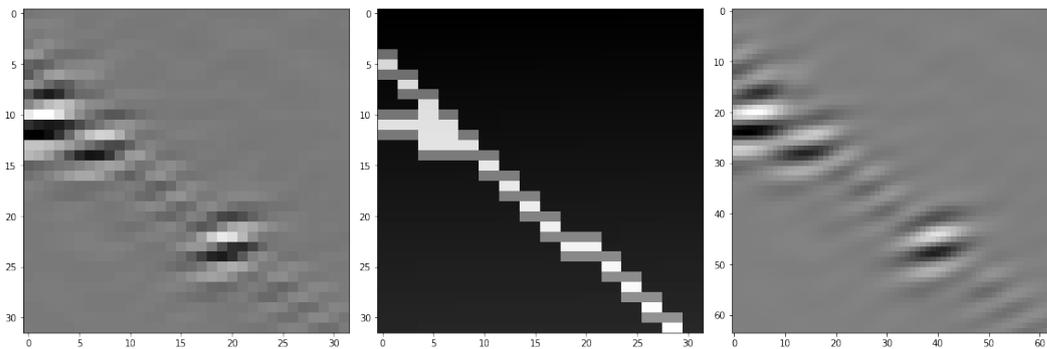


Figura 4.5: Imagem em baixa resolução, mapa de velocidade e imagem em alta resolução

Por último, os dados são separados em três partições, a partição de treino, de teste e de validação. Esta separação é feita de forma randômica, com uma proporção de 50% para treino, 30% para teste e 20% para validação.

Ao final deste procedimento, obtém-se um total de 2688 recortes. Sendo estes equivalentes a 1344 no treino, 516 na validação e 828 no teste.

Total	1860
Média	0.0246
Std	0.0413
Mínimo	0.0000
25%	0.0011
50%	0.0083
75%	0.0288
Máximo	0.2589

Tabela 4.1: Variabilidade por recorte

4.1.1

Análise exploratória

Diante do que foi apresentado, uma análise exploratória é feita em cima dos recortes de treino e validação. Inicialmente, foi examinada a variabilidade das informações dos recortes, visto que exemplos onde a imagem permanece constante não se mostram relevantes para o aprendizado do modelo. A Tabela 4.1.1 apresenta os resultados desta análise e o histograma da variabilidade é ilustrado na Figura 4.6.

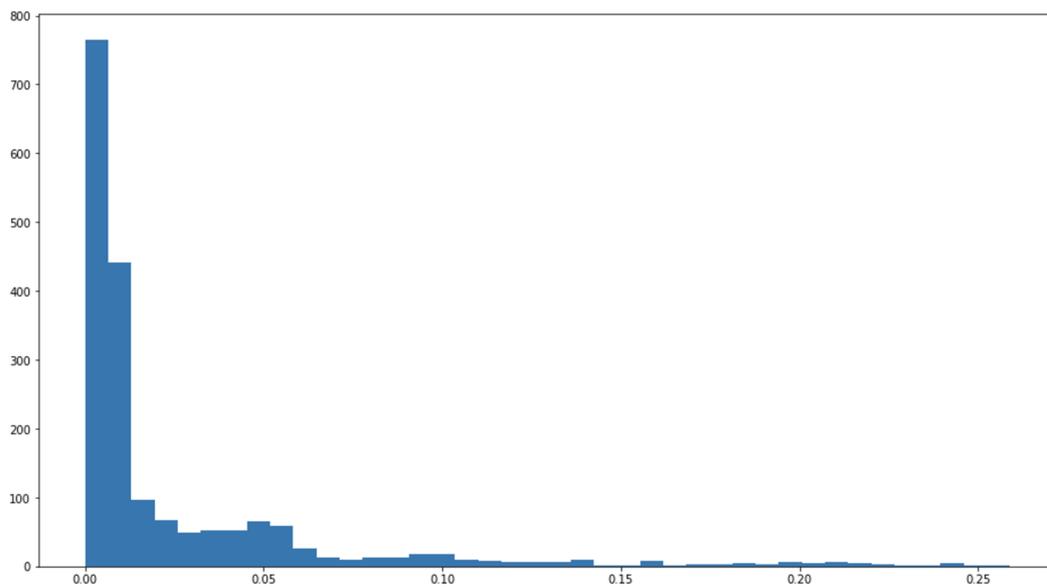


Figura 4.6: Histograma da variabilidade por recorte

Os resultados apresentados mostram uma alta taxa de recortes não informativos (Figura 4.7). Um total de 25% (368 recortes) dos dados da imagem de dimensão 1040 x 7760 apresentam desvio padrão 0. Sendo este percentual equivalente a 15.79% (192 recortes) no reservatório de dimensão 1216 x 6912. Uma análise mais cuidadosa em cima dos reservatórios mostra que este conjunto corresponde a região referente ao mar nas imagens. Visto isto, estas regiões foram removidas dos conjuntos de treino, validação e teste.

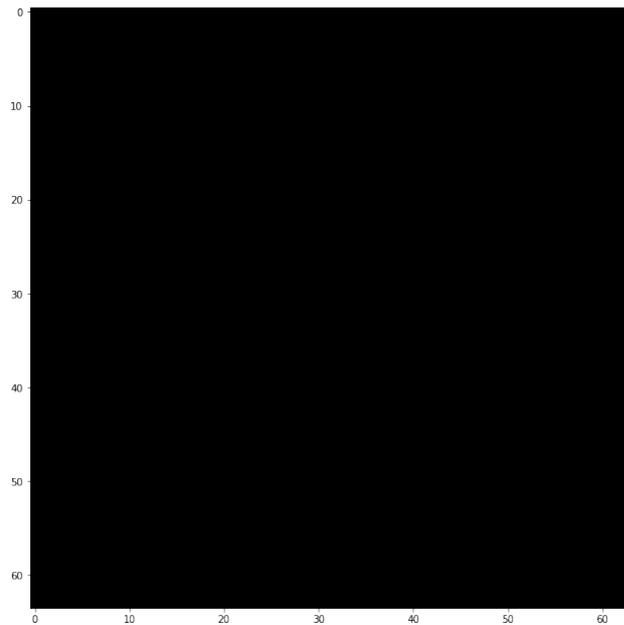
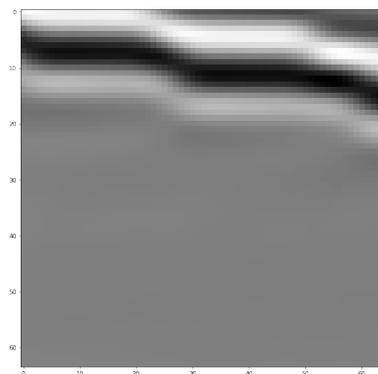
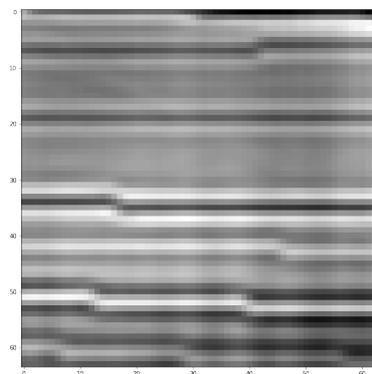


Figura 4.7: Recorte com desvio padrão 0

As Figuras 4.8(a) e 4.8(b) ilustram respectivamente exemplos de contextos de desvio padrão médio e desvio padrão máximo.



4.8(a): Recorte com desvio padrão médio



4.8(b): Recorte com desvio padrão máximo

Figura 4.8: Exemplos de recortes

4.1.2

Data augmentation

Com o intuito de contornar o problema da escassa quantidade de dados, principalmente após a redução de cerca de 21% do *dataset* devido a questão da faixa de mar, foi empregada uma técnica bastante comum de *data augmentation*: o giro horizontal das imagens. Com isto, o conjunto de dados de treino foi ampliado em duas vezes.

Em suma, o *dataset* sintético, fornecido pela Petrobras, resultaram em um conjunto com 3192 recortes. Sendo estes separados em 2128 para o treino, 425 para a validação e 639 para o teste.

4.1.3 F3

O F3 [45] é um *dataset* público amplamente utilizado por pesquisadores na área de sísmica. Ele corresponde a dados sísmicos 3D de um bloco de 384km² no setor holandês do Mar do Norte (Figura 4.9). Este conjunto de dados é composto por 651 *inlines*, dados na direção das linhas sísmicas, e 951 *crosslines*, na direção perpendicular ao *inline*, com intervalo de amostragem equivalente a 4ms. A Figura 4.10 apresenta uma ilustração do *inline* 532.



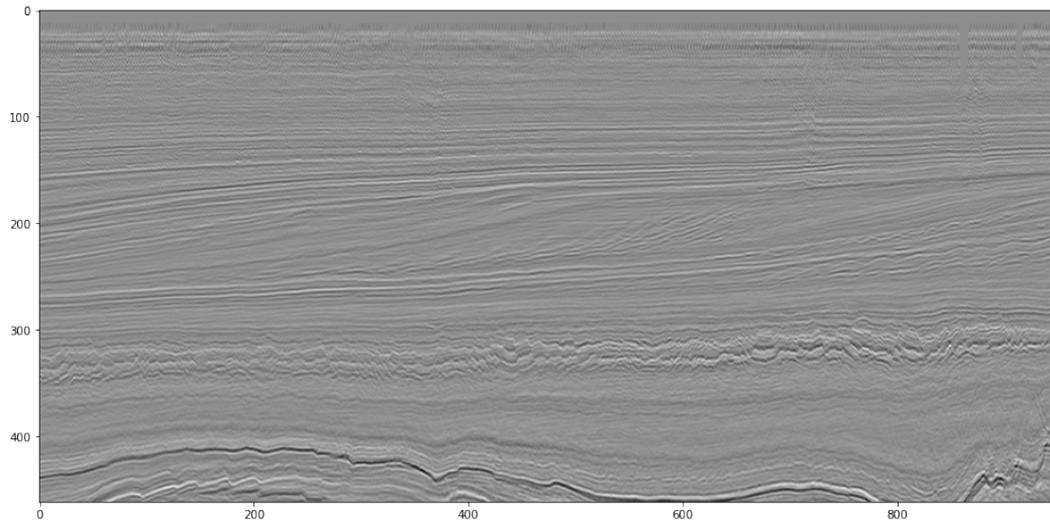
Figura 4.9: Representação da localização do F3

A adoção do F3, para a elaboração deste projeto, aumenta drasticamente o número de contextos disponíveis para treinamento e avaliação dos modelos. Seguindo o procedimento de construção do *dataset* de SR, obtemos um total de 63798 recortes, distribuídos entre 31899 no treino, 12760 na validação e 19139 no teste. A Figura 4.11 apresenta um exemplo de recorte gerado.

4.2 Baseline

Seguindo as estratégias, comumente empregadas nos trabalhos de visão computacional, foi escolhido o algoritmo de interpolação bicúbica para a definição do *baseline*.

A interpolação bicúbica utiliza a média ponderada dos 16 *pixels* vizinhos que se encontram ao redor do *pixel* central. Neste algoritmo, cada pixel vizinho

Figura 4.10: *Inline 532*

recebe um peso correspondente à sua distância geométrica ao pixel central. Desta forma, a importância dos vizinhos é dada pelo uso de pesos que são baseados na *spline* cúbica. O referido algoritmo de interpolação bicúbica é apresentado na Figura 4.12.

4.3

Modelos

Tendo em vista que o objetivo deste trabalho é a aplicação de modelos de importância cronológica da literatura de SR de imagens naturais em dados sísmicos, foram empregados os métodos propostos no capítulo referente a fundamentação teórica. Portanto, considerando as particularidades e quantidade disponível dos dados sísmicos, as arquiteturas propostas pelos artigos selecionados foram alteradas com base em experimentações.

O modelo introdutório das CNNs na super resolução, o SRCNN, assim como o *Residual Dense Network*, devido a simplicidade de codificação, foram implementados para este trabalho utilizando o *framework* Tensorflow [34] para desenvolvimento de redes neurais.

Devido a singularidade do método em *Deep image prior*, será empregado o código disponibilizado no github ¹.

Em relação ao SAN, foram implementadas soluções para a melhora da velocidade da computação da normalização de covariância. Além disso, foram implementados blocos de redes neurais de maior complexidade. Dito isto, optou-se pela utilização do código disponibilizado no github ².

¹https://dmitryulyanov.github.io/deep_image_prior

²<https://github.com/daitao/SAN>

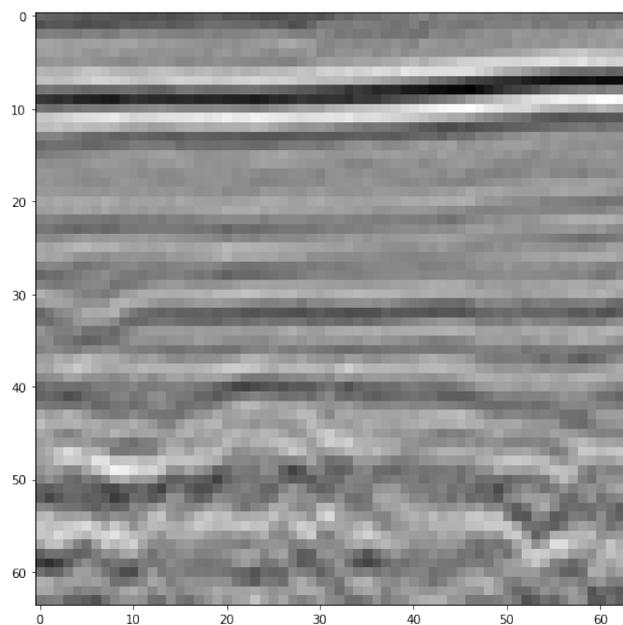


Figura 4.11: Exemplo de recorte obtido no F3

Ademais, com o intuito de avaliar os modelos apresentados na literatura de super resolução na sísmica, as arquiteturas e propostas apresentadas no capítulo referente a trabalhos relacionados foram implementadas utilizando novamente o *framework* Tensorflow [34].

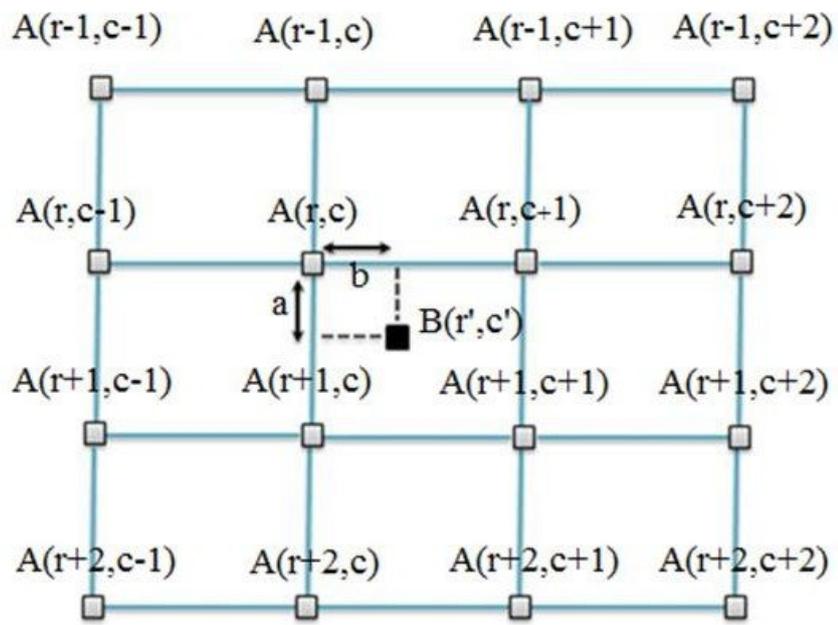


Figura 4.12: Interpolação bicúbica retirada de [23]

5 Experimentação

Este capítulo apresenta a avaliação do modelo e é estruturada como segue: A Seção 5.1 apresenta a configuração da experimentação. Em seguida, a Seção 5.2 apresenta os resultados obtidos.

5.1 Configuração

Cada modelo possui suas configurações específicas, as quais foram definidas a partir de experimentações e da análise das arquiteturas propostas pelos artigos correspondentes. Entretanto, algumas configurações foram definidas como padrões e são adotadas em todas as experimentações. São estas a utilização do otimizador Adam, *batch* de tamanho 64 e a adoção de uma política de *early stopping*.

Em relação ao *early stopping*, foi empregada a estratégia de parar o treinamento do modelo caso em uma quantidade n de épocas, a qual denominamos paciência, não seja atingida uma melhora na *loss* da validação do modelo superior a um dado valor de tolerância.

As experimentações, realizadas para definição das arquiteturas e parâmetros adotados pelos modelos, se baseiam no *dataset* Petrobras. O F3 é adotado para obtenção dos resultados, onde os modelos serão treinados com o novo conjunto de dados. Os únicos hiperparâmetros a serem alterados são a paciência e o número de épocas, visto que o número de atualizações do gradiente por época aumenta proporcionalmente ao tamanho do *dataset*.

Todos os experimentos relatados utilizaram a plataforma Google Colab¹ e foram realizados com o uso de GPU. Devido a natureza do ambiente utilizado, a GPU disponível para treinamento dos modelos variava entre a T4 e a P100 conforme a disponibilidade do serviço.

5.1.1 SRCNN

Devido à simplicidade da arquitetura proposta pelo artigo, a estrutura do modelo se manteve a mesma à apresentada pelos autores.

¹<https://research.google.com/colaboratory/>

Durante o treinamento, a *learning rate* utilizada foi 0,0001. O modelo foi treinado em um total de 5.000 épocas e foi empregado uma paciência de 15 épocas com tolerância de 10^{-5} .

5.1.2 RDN

Em seu artigo, os autores da RDN experimentam com a arquitetura, alterando a quantidade de blocos RDB, assim como a quantidade de camadas convolutivas presentes no bloco. Ao final do artigo, o número de blocos RDB proposto foi $D = 20$, e o número de convoluções por blocos foi de $C = 6$. Entretanto, estes valores quando testados não apresentaram boa performance no *dataset* fornecido pela Petrobras.

Em virtude deste fato, foram feitas experimentações de forma gulosa em relação ao C e D . Para este propósito, foram treinadas durante 500 épocas variações da arquitetura. O objetivo era definir os números de blocos RDB e suas camadas convolutivas com base no PSNR obtido por seus melhores modelos.

Como ilustrado na Figura 5.1, a evolução da métrica avaliada apresenta alta instabilidade ao longo do treinamento dos modelos. Entretanto, é possível observar uma melhor performance da curva azul, correspondente a rede com $C = 4$ e $D = 6$, nas 100 primeiras épocas. Mas, como apresentado na Tabela 5.1, o PSNR correspondente a estes parâmetros tem seu aprendizado estabilizado abaixo dos demais. Com isto, os parâmetros definidos para o contexto deste projeto são $C = 8$ e $D = 6$.

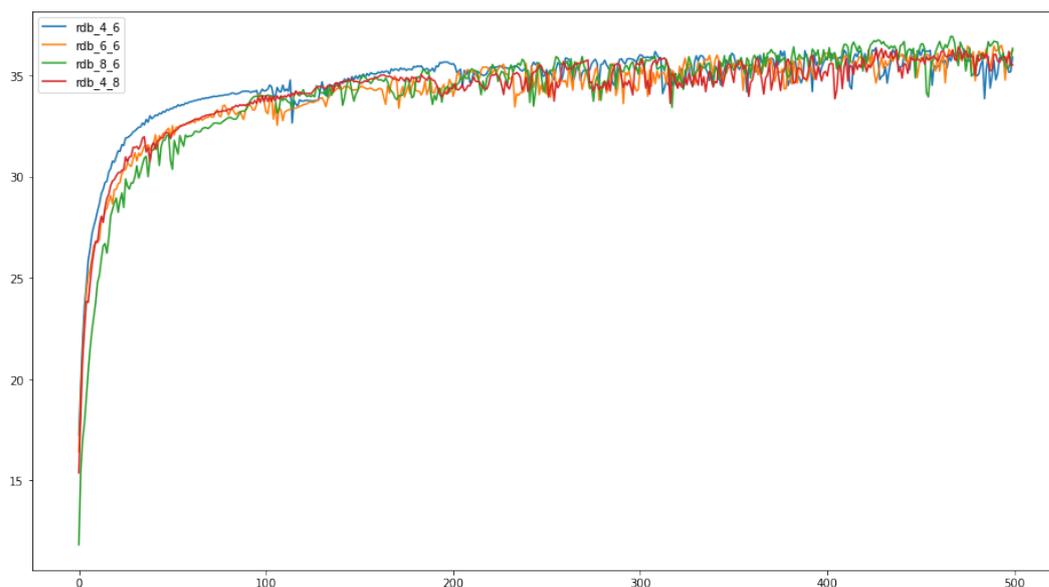


Figura 5.1: Comparativo de PSNR por arquitetura da RDN

C	D	PSNR Val
4	6	36.38
6	6	36.50
8	6	36.94
4	8	36.42

Tabela 5.1: Melhor modelo por arquitetura da RDN

Em relação ao treinamento, a *learning rate* utilizada foi de 0,0001. O modelo foi treinado em um total de 5.000 épocas e foi empregado uma paciência de 25 épocas com tolerância de 10^{-4} .

5.1.3 DIP

Em relação ao modelo apresentado pelo DIP, sua configuração consiste na definição dos parâmetros de número de iterações, o desvio padrão do ruído uniforme e a rede neural a ser otimizada.

O número de iterações recomendado para uma super resolução de fator de ampliação de 8 vezes é 4.000, para o fator 4 é 2.000 e para um fator de 2 vezes os autores não fizeram experimentações.

No que se refere ao desvio padrão do ruído, os valores recomendados para fatores 4 e 8 são respectivamente 0,3 e 0,5.

Visando definir o valor de número de iterações e o desvio padrão, foi feita uma busca gulosa em cima de ambos os parâmetros com o intuito de obter o melhor PSNR. Ao fim desta etapa, foi definido como número de iterações e desvio padrão 2.000 e 0,5.

A biblioteca disponibilizada do DIP apresenta 3 tipos diferentes de redes neurais para serem usadas: a "skip" que consiste em uma arquitetura *enconder-decoder* com *skip connections*, a "UNet" e a "ResNet". Visando a escolha do melhor modelo, a performance das 3 redes foi avaliada. Como mostrado na Tabela 5.2, a "ResNet" obteve o melhor resultado.

Modelo	PSNR Val
Skip	34.38
ResNet	34.80
UNet	34.79

Tabela 5.2: PSNR para diferentes arquiteturas adotadas no DIP

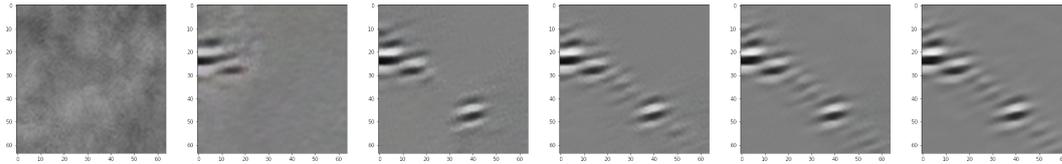


Figura 5.2: Evolução da otimização do DIP

5.1.4 SAN

Para a experimentação com o SAN, devido a empecilhos com a biblioteca, a adoção do ambiente do Google Colab se mostrou inconveniente. Com isto, optou-se pela utilização de um ambiente Linux local. Por consequência, a GPU utilizada neste caso foi uma GTX 1060.

Os parâmetros disponibilizados para customização são o número de blocos LSRAG($n_resgroups$) e o número de blocos residuais($n_resblocks$). Devido a limitações técnicas, a arquitetura com os valores sugeridos de 20 blocos LSRAG e 10 blocos residuais não foi suportada na memória do CUDA. Com isto, foi adotado $n_resgroups = 4$ e $n_resblocks = 6$.

Durante o treinamento, a *learning rate* foi inicializada como 0,0001 e teve seu valor reduzido pela metade a cada 200 épocas. O modelo foi treinado em um total de 600 épocas e não foi empregado nenhum mecanismo de *early stopping*.

5.1.5 M-RUDSRv2

A arquitetura apresentada no artigo do M-RUDSR apresenta como entrada contextos de tamanho 256 x 256. Mas, os nossos contextos foram definidos com um tamanho 4 vezes menor. Isto por sua vez afeta diretamente na profundidade da arquitetura proposta baseada em U-net, uma vez que a cada camada de *downsample*, o tamanho da entrada é reduzido pela metade. Isto, por sua vez, fez com que a arquitetura fosse alterada. Logo, foi utilizada uma camada de *downsample* a menos. Além disso, os 35 blocos densos apresentados na rede estavam causando *overfit*. Para contornar este problema, este valor foi reduzido para 16.

Em relação ao treinamento, a *learning rate* utilizada foi de 0,0001. O modelo foi treinado em no máximo 5.000 épocas e foi empregado uma paciência de 25 épocas com tolerância de 10^5 .

Modelo	Sismograma		Sismograma + Velocidade	
	PSNR Val	PSNR Teste	PSNR Val	PSNR Teste
Baseline	30.47	30.62	-	-
SRCNN	33.97	33.96	33.95	33.80
RDN	38.71	38.51	38.63	38.33
DIP	34.67	34.88	-	-
SAN	38.83	38.62	38.70	38.51
M-RUDSRv2	37.95	38.24	37.99	38.05
Li [44]	37.54	37.61	37.50	37.53

Tabela 5.3: Avaliação do PSNR dos modelos no conjunto de validação e teste do *dataset* Petrobras

5.1.6

Li

Assim como na M-RUDSR, a matriz de entrada da rede possui tamanho 4 vezes maior do que as entradas preparadas no *dataset* Petrobras. Assim, a mesma abordagem de remover uma camada de *downsample* foi aplicada. Fora isto, o modelo manteve a mesma arquitetura apresentada no artigo.

A *learning rate* adotada no treino foi de 0,0001. O modelo foi treinado em no máximo 5.000 épocas e foi empregado uma paciência de 25 épocas com tolerância de 10^4 .

5.2

Resultados

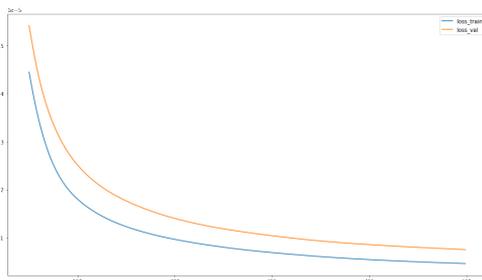
Os resultados aqui apresentados foram avaliados em três cenários diferentes. O primeiro é a utilização exclusiva das imagens de reservatórios fornecidos pela Petrobras. O segundo, baseado nos resultados apresentados por [44], que mostram um ganho de performance ao se adicionar dados sísmicos adicionais, utiliza as imagens de reservatórios em conjunto aos seus modelos de velocidade. Por último, o terceiro, é a avaliação dos modelos e arquiteturas propostas no *dataset* F3.

Vale ressaltar que, no segundo cenário, todos os modelos treináveis tiveram apenas a camada de entrada alterada para receberem entradas com 2 canais.

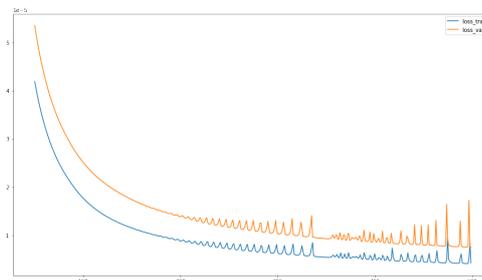
Como apresentado na Tabela 5.3, os resultados das redes neurais mostram uma significativa melhora no PSNR em relação à interpolação bicúbica. Ademais, as arquiteturas apresentadas nos trabalhos relacionados também foram superados pelo RDN e pelo SAN, sendo que este último obteve o melhor resultado. Este fato demonstra o potencial que a literatura de super resolução de imagens naturais traz ao domínio da sísmica.

Modelo	F3	
	PSNR Val	PSNR Teste
Baseline	24.04	24.00
SRCNN	25.10	25.12
RDN	26.97	30.00
DIP	24.35	24.27
SAN	30.12	30.10
M-RUDSRv2	26.33	26.34
Li [44]	26.03	25.99

Tabela 5.4: Avaliação do PSNR dos modelos no conjunto de validação e teste do F3



5.3(a): Treinamento sismograma



5.3(b): Treinamento sismograma + velocidade

Figura 5.3: Instabilidade no treinamento dos modelos ao adicionar dados sísmicos extras

Diferente do esperado, a adição dos dados de velocidade trouxeram ganho de performance apenas no conjunto de validação do M-RUDSRv2, sendo este ganho de 0,04. Isto indica que estes dados sintéticos gerados e fornecidos para o contexto deste projeto não são relevantes como os reunidos na pesquisa do artigo do M-RUDSRv2.

Além do mais, a adição destes dados aumentou a instabilidade da *loss* durante o treinamento do modelo. Um exemplo claro disto pode ser visto no treinamento do SRCNN, como ilustrado na Figura 5.2

Ao comparar os resultados dos dois *datasets* adotados por este projeto, percebe-se que a performance do DIP foi prejudicada quando aplicado no F3. Esta diferença pode ter sido obtida devido à presença de ruídos, provenientes da captura dos dados, presente em conjuntos de dados reais. Entretanto, em ambos os casos, o DIP trouxe resultados superiores à interpolação bicúbica, o que, de acordo com os objetivos iniciais de seu artigo, validam sua abordagem.

A Figura 5.4 apresenta o resultado da saída dos modelos para um único contexto do *dataset* Petrobras. Uma avaliação visual destas imagens mostra uma pequena variação em relação à imagem original no canto inferior direito da imagem interpolada. Ademais, é observada uma grande semelhança entre

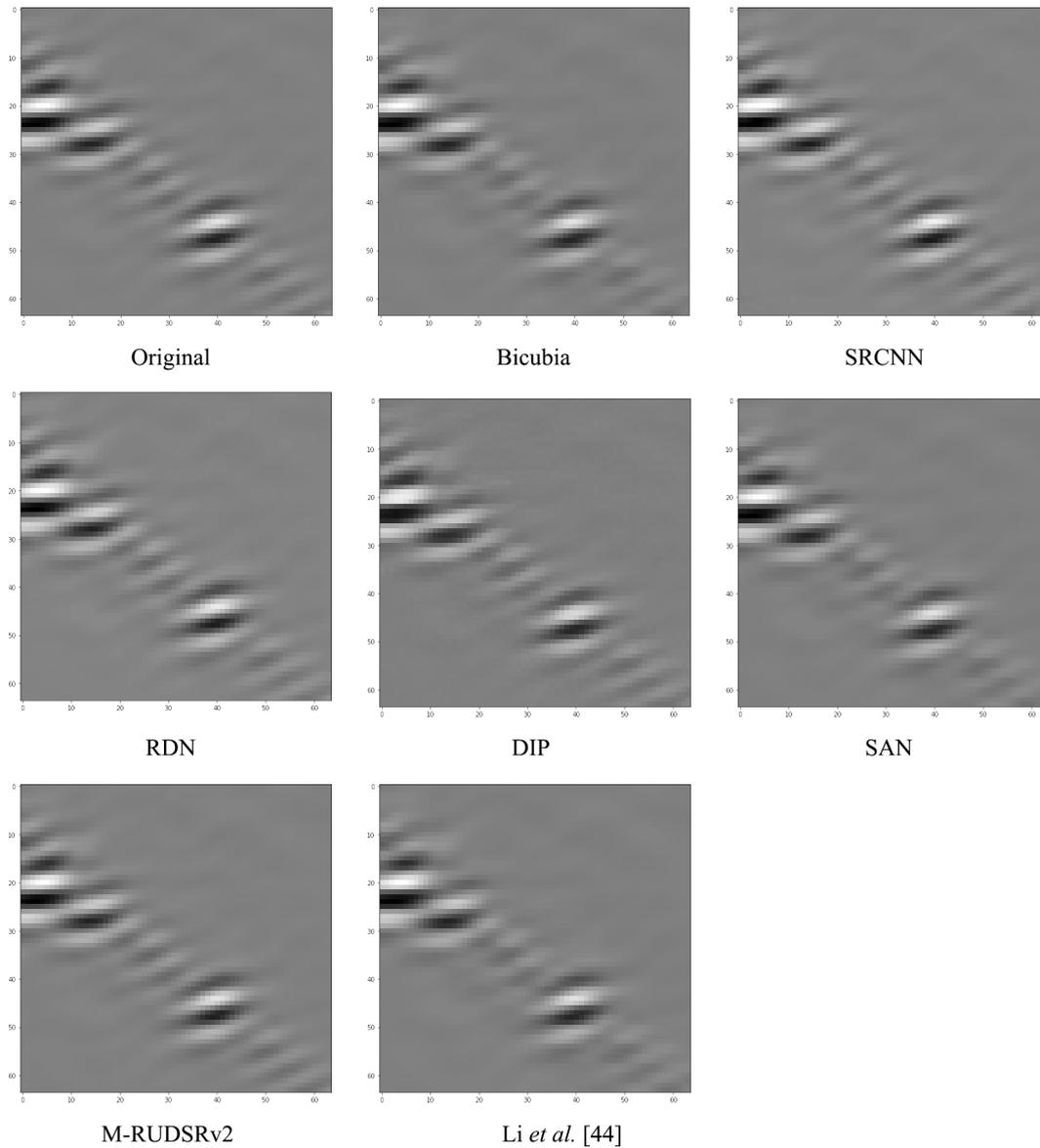


Figura 5.4: Exemplo de saída gerada por cada modelo no *dataset* Petrobras

todos os modelos propostos.

Por sua vez, a Figura 5.5 apresenta o resultado da saída dos modelos no F3. Diferente do exemplo anterior, as diferenças entre a qualidade dos resultados estão claramente visíveis, principalmente se observada a imagem gerada pela interpolação e pelo DIP. Todas as imagens apresentadas estão desprovidas dos detalhes presentes na imagem original de alta resolução.

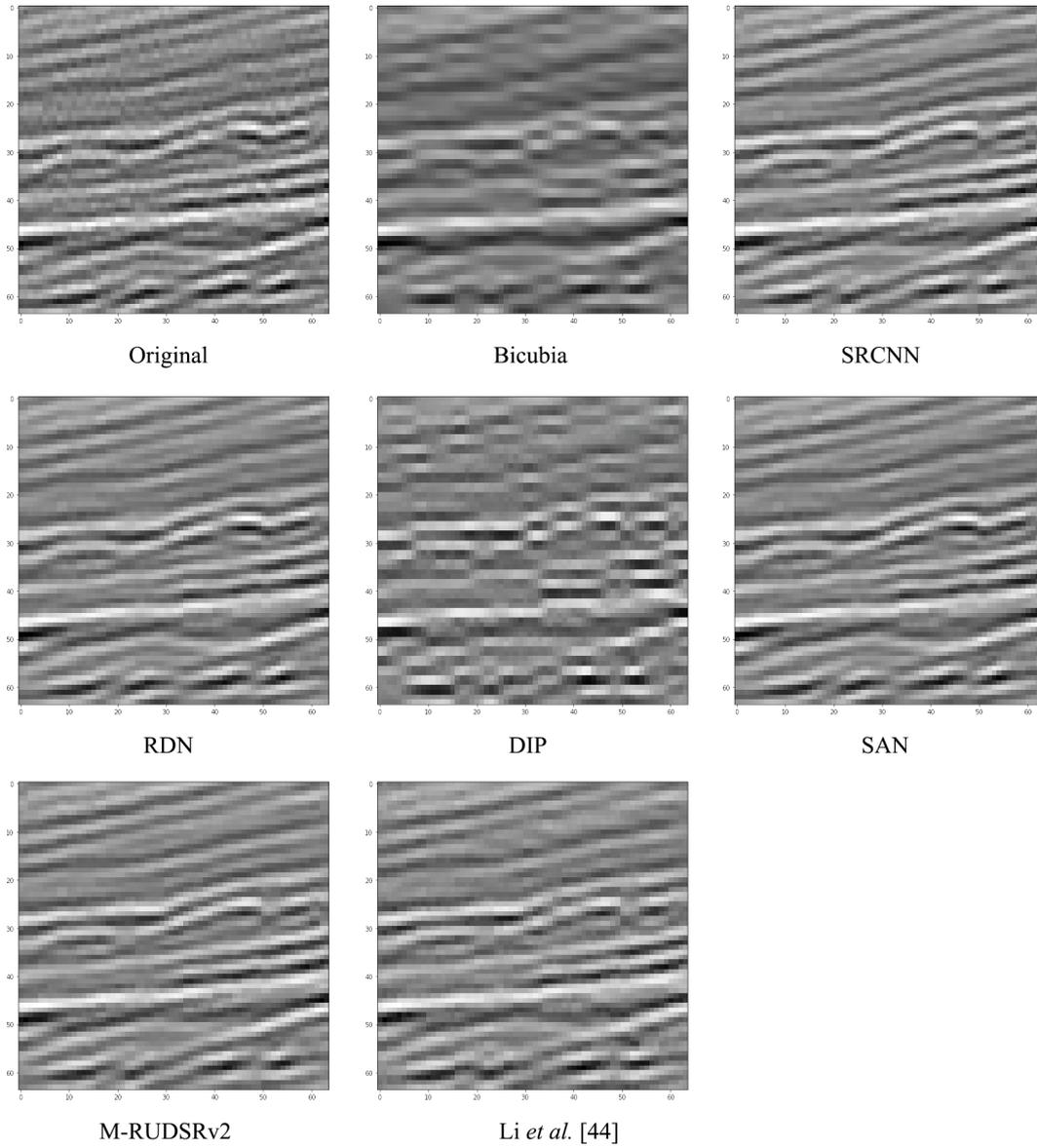


Figura 5.5: Exemplo de saída gerada por cada modelo no F3

6

Considerações Finais

Com o intuito de apresentar os ganhos que a utilização das redes propostas pela literatura de super resolução de imagens naturais podem trazer ao domínio da sísmica, foram utilizados 4 modelos que adotam técnicas que se mostraram notáveis nos avanços das pesquisas sobre redes neurais ao tempo de suas publicações.

Uma análise em cima dos resultados apresentados demonstram que a utilização dos modelos mais avançados de SR de imagens naturais trazem um ganho no PSNR de 0,38 na *dataset* sintético da Petrobras e 0,76 nos dados públicos do *dataset* F3.

O primeiro modelo, o SRCNN, introduz pela primeira vez redes neurais na tarefa de SR, superando todos os demais métodos existentes na época. No contexto deste trabalho, o SRCNN se mostrou superior a interpolação apresentada como *baseline*.

O RDN, o segundo modelo em ordem cronológica, emprega blocos residuais densos para fazer pleno uso das *features* hierárquicas da imagem em baixa resolução. A utilização dos blocos RDB se mostraram de grande valor para um aumento da métrica de PSNR. Com isto, o RDN obteve o segundo melhor resultado em ambos os *datasets* empregados.

O terceiro trabalho avaliado, o *deep image prior*, apresenta uma abordagem onde não há treinamento da rede neural. Ele parte do princípio que a estrutura de uma rede geradora é suficiente para capturar uma vasta quantidade de estatísticas da imagem e identificar a informação perdida entre a baixa e alta resolução. Analisando os resultados obtidos neste trabalho, é observado que o DIP trouxe resultados superiores à interpolação bicúbica, mostrando que esta abordagem possui relevância no domínio da sísmica.

Por último, o *Second-Order Attention Network* aplica mecanismos de atenção trazendo uma melhoria no uso e no aprendizado da correlação das *features*. Esta rede apresenta os melhores resultados obtidos nos *datasets* da sísmica empregados neste trabalho.

Durante o início da pesquisa deste trabalho, o SAN apresentava os resultados estado da arte para parte dos *datasets* de *benchmark* da tarefa de SR de imagens naturais. Com isto, foi o último modelo da literatura a ser

empregado.

Para validar os ganhos obtidos por estes modelos da literatura de SR de imagens naturais, foram avaliados 2 arquiteturas propostas em trabalhos focados em dados sísmicos. Ambas arquiteturas apresentam estruturas bastante similares baseadas na U-net. O M-RUDSRv2 apresentou o melhor resultado quando comparado com a arquitetura de Li *et al.* [43]. Ademais, suas métricas ficaram bem próximas às do RDN.

O M-RUDSRv2 trouxe toda uma nova gama de experimentações ao apresentar os ganhos que o utilização de dados sísmicos adicionais podem trazer na tarefa. Entretanto, foi verificado que a adição dos dados de velocidade fornecidos para o projeto não traz ganhos como no caso dos dados utilizados no artigo [44].

6.1

Trabalhos futuros

Uma particularidade dos dados sísmicos em relação a imagens naturais é que o primeiro trata de sinais. Tendo isto em vista, trabalhos que lidam diretamente com sinais podem ser de grande relevância para o avanço dos modelos de SR de dados sísmicos. Temos como exemplos o *neural implicit representation* e uma ampla bibliografia de trabalhos voltados para imagens médicas.

Além do mais, uma aplicação dos modelos propostos em diferentes aumentos de resolução e em outros tipos de reservatórios e dados sísmicos podem trazer novos *insights* e demonstrar uma maior robustez e confiança em relação a seus resultados.

Referências bibliográficas

- [1] ZHANG, Yulun et al. Residual dense network for image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018. p. 2472-2481.
- [2] LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. *nature*, v. 521, n. 7553, p. 436-444, 2015.
- [3] KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, v. 60, n. 6, p. 84-90, 2017.
- [4] SHI, Wenzhe et al. Cardiac image super-resolution with global correspondence using multi-atlas patchmatch. In: International conference on medical image computing and computer-assisted intervention. Springer, Berlin, Heidelberg, 2013. p. 9-16.
- [5] ZOU, Wilman WW; YUEN, Pong C. Very low resolution face recognition problem. *IEEE Transactions on image processing*, v. 21, n. 1, p. 327-340, 2011.
- [6] HORE, Alain; ZIOU, Djemel. Image quality metrics: PSNR vs. SSIM. In: 2010 20th international conference on pattern recognition. IEEE, 2010. p. 2366-2369.
- [7] WANG, Zhou; SIMONCELLI, Eero P.; BOVIK, Alan C. Multiscale structural similarity for image quality assessment. In: The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003. Ieee, 2003. p. 1398-1402.
- [8] ULYANOV, Dmitry; VEDALDI, Andrea; LEMPITSKY, Victor. Deep image prior. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018. p. 9446-9454.
- [9] DAI, Tao et al. Second-order attention network for single image super-resolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019. p. 11065-11074.

- [10] ZHOU, Fei; YANG, Wenming; LIAO, Qingmin. Interpolation-based image super-resolution using multisurface fitting. *IEEE Transactions on Image Processing*, v. 21, n. 7, p. 3312-3318, 2012.
- [11] DONG, Chao et al. Learning a deep convolutional network for image super-resolution. In: *European conference on computer vision*. Springer, Cham, 2014. p. 184-199.
- [12] IRANI, Michal; PELEG, Shmuel. Improving resolution by image registration. *CVGIP: Graphical models and image processing*, v. 53, n. 3, p. 231-239, 1991.
- [13] BEVILACQUA, Marco et al. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012.
- [14] LECUN, Yann et al. Backpropagation applied to handwritten zip code recognition. *Neural computation*, v. 1, n. 4, p. 541-551, 1989.
- [15] ZEYDE, Roman; ELAD, Michael; PROTTER, Matan. On single image scale-up using sparse-representations. In: *International conference on curves and surfaces*. Springer, Berlin, Heidelberg, 2010. p. 711-730.
- [16] LIANG, Jingyun et al. Swinir: Image restoration using swin transformer. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021. p. 1833-1844.
- [17] DENG, Jia et al. Imagenet: A large-scale hierarchical image database. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009. p. 248-255.
- [18] BUSSON, Antonio José G. et al. Seismic shot gather noise localization using a multi-scale feature-fusion-based neural network. *arXiv preprint arXiv:2005.03626*, 2020.
- [19] BUCKER, Eduardo Betine et al. A Deep Convolutional Network for Seismic Shot-Gather Image Quality Classification. *arXiv preprint arXiv:1912.01148*, 2019.
- [20] Colcher, Sérgio. "Redes Convolucionais aplicadas à Classificação de Ruído Sísmico." *MsC diss.*, PUC-Rio, 2020.
- [21] HENRIQUES, Luis Felipe et al. Generating Data Augmentation samples for Semantic Segmentation of Salt Bodies in a Synthetic Seismic Image Dataset. *arXiv preprint arXiv:2106.08269*, 2021.

- [22] Cabral Torres, Mateus, Colcher, Sérgio. "Redes Convolucionais aplicadas à Segmentação Semântica de Imagens Sísmicas." MsC diss., PUC-Rio, 2021.
- [23] HISHAM, M. B. et al. An analysis of performance for commonly used interpolation method. *Advanced Science Letters*, v. 23, n. 6, p. 5147-5150, 2017.
- [24] HE, Kaiming et al. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. p. 770-778.
- [25] MARTIN, Gary S.; WILEY, Robert; MARFURT, Kurt J. Marmousi2: An elastic upgrade for Marmousi. *The leading edge*, v. 25, n. 2, p. 156-166, 2006.
- [26] LEDIG, Christian et al. Photo-realistic single image super-resolution using a generative adversarial network. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017. p. 4681-4690.
- [27] LIM, Bee et al. Enhanced deep residual networks for single image super-resolution. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2017. p. 136-144.
- [28] TAI, Ying et al. Memnet: A persistent memory network for image restoration. In: *Proceedings of the IEEE international conference on computer vision*. 2017. p. 4539-4547.
- [29] SHI, Wenzhe et al. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. p. 1874-1883.
- [30] KINGMA, Diederik P.; BA, Jimmy. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [31] MARTIN, David et al. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001. IEEE*, 2001. p. 416-423.
- [32] HUANG, Jia-Bin; SINGH, Abhishek; AHUJA, Narendra. Single image super-resolution from transformed self-exemplars. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015. p. 5197-5206.

- [33] MATSUI, Yusuke et al. Sketch-based manga retrieval using manga109 dataset. *Multimedia Tools and Applications*, v. 76, n. 20, p. 21811-21838, 2017.
- [34] ABADI, Martín et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.
- [35] RONNEBERGER, Olaf; FISCHER, Philipp; BROX, Thomas. U-net: Convolutional networks for biomedical image segmentation. In: *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015. p. 234-241.
- [36] <https://towardsdatascience.com/demystifying-deep-image-prior-7076e777e5ba> , acessado dia 2 de fevereiro de 2022.
- [37] HU, Jie; SHEN, Li; SUN, Gang. Squeeze-and-excitation networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018. p. 7132-7141.
- [38] ZHANG, Yulun et al. Image super-resolution using very deep residual channel attention networks. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018. p. 286-301.
- [39] WANG, Xiaolong et al. Non-local neural networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018. p. 7794-7803.
- [40] LI, Peihua et al. Is second-order information helpful for large-scale visual recognition?. In: *Proceedings of the IEEE international conference on computer vision*. 2017. p. 2070-2078.
- [41] SHI, Wenzhe et al. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. p. 1874-1883.
- [42] LI, Yinshuo et al. Multitask learning for super-resolution of seismic velocity model. *IEEE Transactions on Geoscience and Remote Sensing*, v. 59, n. 9, p. 8022-8033, 2020.
- [43] LI, Yinshuo et al. Super-Resolution of Seismic Velocity Model Guided by Seismic Data. *IEEE Transactions on Geoscience and Remote Sensing*, v. 60, p. 1-12, 2021.

- [44] LI, Jintao; WU, Xinming; HU, Zhanxuan. Deep learning for simultaneous seismic image super-resolution and denoising. *IEEE Transactions on Geoscience and Remote Sensing*, v. 60, p. 1-11, 2021.
- [45] OSR; DGB EARTH SCIENCES. Netherlands offshore f3 block-survey. 2017.