

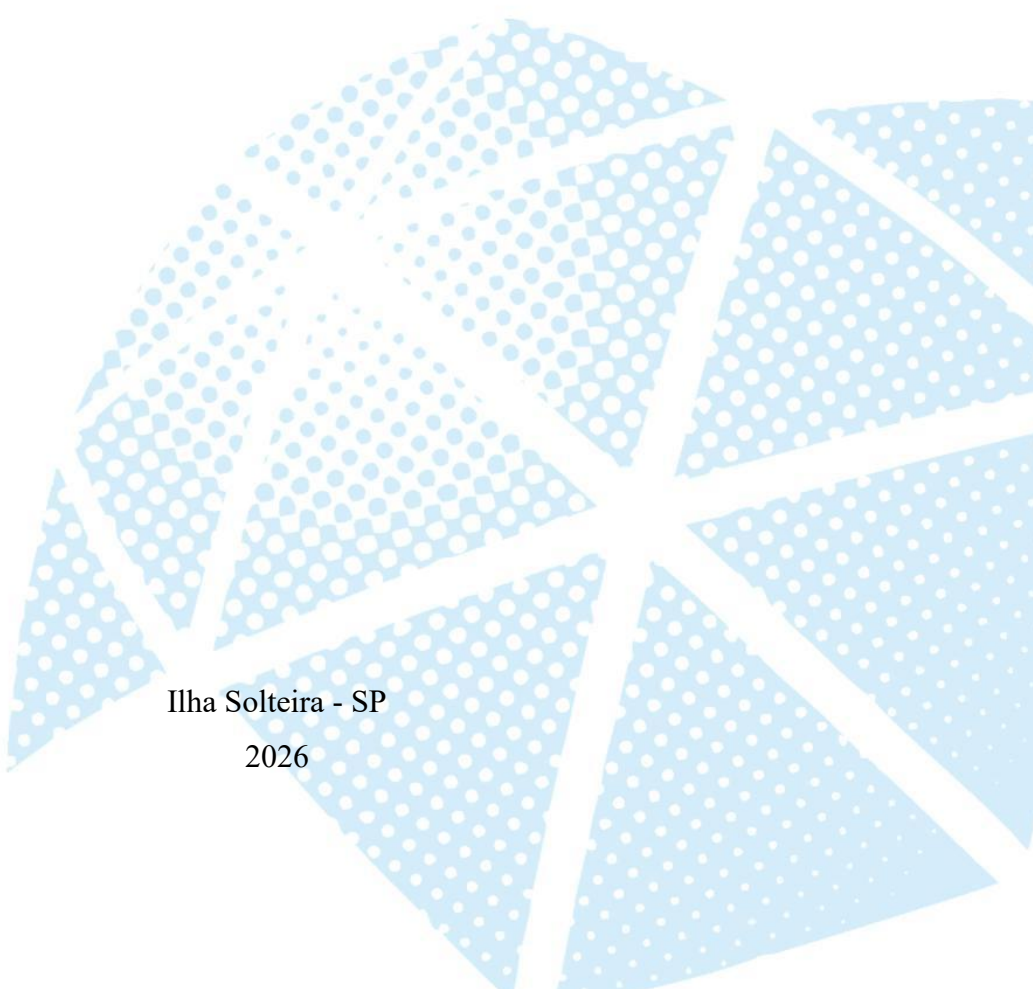
**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
CAMPUS DE ILHA SOLTEIRA**

RONALDO GORGULHO FIGUEIREDO OLIVEIRA DOS SANTOS

**ANÁLISE DO USO DE APRENDIZADO DE MÁQUINA INFORMADO POR FÍSICA
PARA SUPRESSÃO DE
RUÍDO SPECKLE EM IMAGENS SAR DO SENTINEL-1 GRD**

Ilha Solteira - SP

2026



RONALDO GORGULHO FIGUEIREDO OLIVEIRA DOS SANTOS

**ANÁLISE DO USO DE APRENDIZADO DE MÁQUINA INFORMADO POR FÍSICA
PARA SUPRESSÃO DE
RUÍDO SPECKLE EM IMAGENS SAR DO SENTINEL-1 GRD**

Trabalho de conclusão de curso apresentado à
Universidade Estadual Paulista (UNESP),
Faculdade de Engenharia de Ilha Solteira,
Ilha Solteira, para
obtenção do título de Grau acadêmico Bacharel
em Engenharia Civil.

Área de Concentração: Conforme consta no
certificado de aprovação

Orientador(a): Prof. Dr. César Gustavo da
Rocha Lima

Ilha Solteira - SP

2026

FICHA CATALOGRÁFICA
Desenvolvida pela Diretoria Técnica de Biblioteca e Documentação

- S237a Santos, Ronaldo Gorgulho Figueiredo Oliveira dos.
Análise do uso de aprendizado de máquina informado por física para supressão de ruído speckle em imagens SAR do Sentinel-1 GRD / Ronaldo Gorgulho Figueiredo Oliveira dos Santos. -- Ilha Solteira: [s.n.], 2026
124 f. : il.
- Trabalho de conclusão de curso (Graduação em Engenharia Civil) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira, 2026
- Orientador: César Gustavo da Rocha Lima
Inclui bibliografia
1. Redes neurais profundas. 2. Sensoriamento remoto orbital. 3. Radar de abertura sintética. 4. Filtragem multitemporal. 5. Ruído multiplicativo.

FOLHA DE APROVAÇÃO

Aluno: RONALDO GORGULHO FIGUEIREDO OLIVEIRA DOS SANTOS

Título: ANÁLISE DO USO DE APRENDIZADO DE MÁQUINA INFORMADO POR FÍSICA PARA SUPRESSÃO DE RUÍDO SPECKLE EM IMAGENS SAR DO SENTINEL-1 GRD

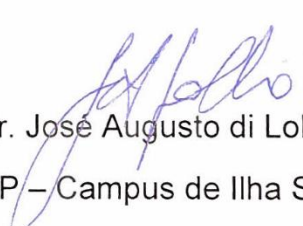
Trabalho de Conclusão de Curso (TCC) apresentado como parte dos requisitos para obtenção do grau de Engenheiro Civil, junto ao Curso de Graduação em Engenharia Civil, da Faculdade de Engenharia da Universidade Estadual Paulista "Júlio de Mesquita Filho", Campus de Ilha Solteira

COMISSÃO EXAMINADORA



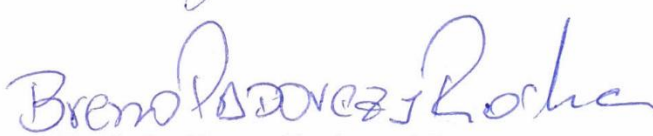
Prof. Dr. César Gustavo da Rocha Lima

FEIS/UNESP – Campus de Ilha Solteira (Orientador)



Prof. Dr. José Augusto di Lollo

FEIS/UNESP – Campus de Ilha Solteira



Prof. Dr. Breno Padovezi Rocha

FEIS/UNESP – Campus de Ilha Solteira

Ilha Solteira

26/02/2026

RESUMO

Este trabalho apresenta o *Multitemporal Anti-Speckle Framework*, um sistema de redes neurais profundas desenvolvido para redução de ruído em dados de Radar de Abertura Sintética da missão Sentinel-1. A metodologia organiza-se em uma estratégia de treinamento híbrida de dois estágios. No primeiro, supervisionado, o modelo aprende a reconstruir a imagem a partir de alvos de referência gerados por combinações multitemporais, utilizando uma função de perda composta que combina termos de fidelidade de pixel e de estrutura. Em seguida, um estágio opcional de refinamento não supervisionado, guiado por propriedades estatísticas do ruído multiplicativo, ajusta os parâmetros do modelo usando apenas imagens com ruído, impondo que o resíduo estimado se aproxime do comportamento teórico do *speckle*. A arquitetura é baseada em U-Net com *backbones* (Swin Transformer e Mamba) e incorpora o módulo de atenção SkipGate, responsável por modular de forma adaptativa a informação que flui pelas conexões de atalho. O *framework* inclui ainda um *pipeline* de aumento de dados com transformações físicas e perceptuais, suporte a treinamento multitarefa e estrutura para estudos de ablação.

Os resultados experimentais, obtidos em *patches* de validação do canal de polarização vertical-vertical, mostram que o *Multitemporal Anti-Speckle Framework* alcança valores de Número Equivalente de Visadas superiores aos de filtros clássicos amplamente utilizados, mantendo a entropia próxima à dos métodos estatísticos e proporcionando fundo mais homogêneo com textura visualmente realista. Embora apresente valores médios de Relação Sinal-Ruído de Pico e Índice de Similaridade Estrutural inferiores aos melhores filtros tradicionais, a variante com perda perceptual oferece um compromisso mais favorável entre suavização e preservação de detalhes, evitando artefatos observados na versão treinada apenas com perda de Erro Médio Absoluto. A etapa de refinamento informado por física melhora a aderência estatística em relação às demais configurações, com reintrodução controlada de flutuações de intensidade. A análise qualitativa, apoiada em comparações visuais e em mapas de atenção, indica que o *framework* consegue concentrar seus recursos em feições estruturalmente relevantes, reforçando a interpretação física de seu funcionamento interno.

De forma geral, o estudo demonstra que a combinação de alvos multitemporais, perda composta e refinamento guiado por estatísticas do ruído permite obter ganhos consistentes de qualidade visual e homogeneidade em relação aos filtros clássicos, preservando informações relevantes para aplicações em geoprocessamento. Ao mesmo tempo, a variabilidade observada em métricas estatísticas mais sensíveis sugere a necessidade de futuras extensões, como incorporação de mapas de incerteza, funções de perda probabilísticas e abordagens adversariais, capazes de tornar a geração de imagens restauradas mais estável e confiável em diferentes contextos de cena.

Palavras-chave: redes neurais profundas; sensoriamento remoto orbital; radar de abertura sintética; filtragem multitemporal; ruído multiplicativo.

ABSTRACT

This work presents Multitemporal Anti-Speckle Framework, a complete deep neural network-based system designed to reduce noise in Synthetic Aperture Radar data from the Sentinel-1 mission. The methodology is organized as a two-stage hybrid training strategy. In the first, supervised stage, the model learns to reconstruct the image from reference targets generated by multitemporal combinations, using a composite loss function that blends pixel-wise and structural fidelity terms. In the second, optional, unsupervised refinement stage, guided by the statistical properties of multiplicative noise, the model parameters are updated using only noisy images, enforcing the residual component to approximate the theoretical behavior of speckle. The architecture is based on a U-Net design with backbones (Swin Transformer and Mamba) and incorporates the SkipGate attention module, which adaptively modulates the information flowing through the skip connections. The framework also includes a data augmentation pipeline with physically and perceptually motivated transformations, support for multitask training, and a structure suitable for ablation studies.

Experimental results on vertical-vertical polarization channel validation patches show that Multitemporal Anti-Speckle Framework achieves higher Equivalent Number of Looks values than widely used classical filters, while keeping Shannon entropy close to that of statistical methods and producing more homogeneous backgrounds with visually realistic texture. Although its average Peak Signal-to-Noise Ratio and Structural Similarity Index Measure values are lower than those of the best traditional filters, the variant trained with a perceptual loss offers a more favorable trade-off between smoothing and detail preservation, avoiding artifacts observed in the model trained with Mean Absolute Error loss alone. The physics-guided refinement stage improves statistical consistency relative to the other deep variants, with controlled reintroduction of intensity fluctuations. Qualitative analysis, supported by visual comparisons and attention maps, indicates that the framework concentrates its modeling capacity on structurally relevant features, reinforcing the physical interpretation of its internal behavior.

Overall, the study demonstrates that combining multitemporal targets, a composite objective function and refinement guided by noise statistics yields consistent gains in visual quality and homogeneity over classical approaches, while preserving information relevant for geospatial applications. At the same time, the variability observed in more sensitive statistical metrics highlights the need for future extensions, such as the incorporation of uncertainty maps, probabilistic loss functions and adversarial schemes, in order to make the generation of restored images more stable and reliable across different scene conditions.

Keywords: deep neural networks; orbital remote sensing; synthetic aperture radar; multitemporal filtering; multiplicative noise.

LISTA DE FIGURAS

Figura 1 – Comparação entre a imagem SAR com speckle e sua média multitemporal.....	21
Figura 2 – Diagrama da arquitetura U-Net.....	27
Figura 3 – Mecanismo de autoatenção baseado em janelas do <i>Swin Transformer</i>	29
Figura 4 – Diagrama da varredura multidirecional para aplicação do <i>Mamba</i> em dados 2D ..	31
Figura 5 – Imagem restaurada com L1 puro vs. perda perceptual composta	34
Figura 6 – Diagrama conceitual do refinamento informado por física.....	35
Figura 7 – Aumentações físicas e perceptuais aplicadas à um <i>patch</i> na polarização VH	43
Figura 8 – Diagrama em árvore da estrutura de diretórios e arquivos principais do MASF	51
Figura 9 – Representação da janela de busca e superfície de correlação	58
Figura 10 – Métricas para o filtro Stroobants na polarização VV	61
Figura 11 – Entropia Local vs. ENL para o filtro Stroobants na polarização VV	61
Figura 12 – Comparação dos filtros espaciais e multitemporais aplicados ao canal VH	62
Figura 13 – Histogramas calculados sem e com a filtragem de pixels inválidos	63
Figura 14 – Histogramas do canal VV no domínio linear e no domínio logarítmico	63
Figura 15 – Área de pixels válidos e inválidos de uma imagem SAR (Banda VH).....	65
Figura 16 – Área de pixels válidos determinada e a “zona segura” calculada	67
Figura 17 – Diagrama do mecanismo <i>SkipGate</i>	71
Figura 18 – Diagrama da arquitetura multitarefa	73
Figura 19 – Histórico de otimização dos hiperparâmetros com Optuna	84
Figura 20 – Relação da função de perda com cada hiperparâmetro	85
Figura 21 – Importância relativa dos hiperparâmetros analisados	86
Figura 22 – Artefatos suprimidos no modelo treinado com perda perceptual.....	90
Figura 23 – Imagem restaurada com modelo treinado vs. modelo refinado	91
Figura 24 – Imagem restaurada com modelo treinado vs. modelo refinado	91
Figura 25 – Comparação qualitativa dos métodos de supressão de ruído	92
Figura 26 – Mapa de atenção do Gate 1	93
Figura 27 – Mapa de atenção do Gate 3	94
Figura 28 – Mapa de atenção do Swin	94
Figura 29 – Mapa de atenção do Mamba	95

LISTA DE TABELAS

Tabela 1 – Análise quantitativa de filtragem na polarização VV	59
Tabela 2 – Análise quantitativa de filtragem na polarização VH.....	59
Tabela 3 – Hiperparâmetros otimizados para diferentes configurações do modelo.....	83
Tabela 4 – Média das métricas na análise quantitativa	87
Tabela 5 – Desvio padrão das métricas na análise quantitativa	89
Tabela 6 – Análise quantitativa dos mapas de atenção	96

LISTA DE QUADROS

Quadro 1 – Descrição das áreas de estudo	54
Quadro 2 – Comparativo entre os estágios de treinamento	76
Quadro 3 – Domínio das métricas de avaliação	80

LISTA DE ABREVIATURAS E SIGLAS

CNN	Rede Neural Convolucional (<i>Convolutional Neural Network</i>)
ENL	Número Equivalente de Visadas (<i>Equivalent Number of Looks</i>)
EPD	Grau de Preservação de Bordas (<i>Edge Preservation Degree</i>)
EPI	Índice de Preservação de Bordas (<i>Edge Preservation Index</i>)
ESA	Agência Espacial Europeia (<i>European Space Agency</i>)
FDA	Função de Distribuição Acumulada
FFT	Transformada Rápida de Fourier (<i>Fast Fourier Transform</i>)
FOM	Figura de Mérito de Pratt (<i>Pratt's Figure of Merit</i>)
GDAL	<i>Geospatial Data Abstraction Library</i>
GPT	Ferramenta de Processamento de Grafos (<i>Graph Processing Tool</i>)
GPU	Unidade de Processamento Gráfico (<i>Graphics Processing Unit</i>)
GRD	<i>Ground Range Detected</i>
InSAR	SAR Interferométrico (<i>Interferometric SAR</i>)
IW	Faixa Interferométrica Ampla (<i>Interferometric Wide Swath</i>)
JSON	<i>JavaScript Object Notation</i>
KL	Kullback-Leibler
KS	Kolmogorov-Smirnov
L1 (LAD)	Desvios Absolutos Mínimos (<i>Least Absolute Deviations</i>)
L2 (LSE)	Erros Quadrados Mínimos (<i>Least Squares Error</i>)
MAP	Estimação a Posteriori Máxima (<i>Maximum A Posteriori</i>)
MASF	<i>Multitemporal Anti-Speckle Framework</i>
MDE	Modelo Digital de Elevação
MLP	Perceptron de Múltiplas Camadas (<i>Multi-Layer Perceptron</i>)
MMSE	Erro Quadrático Médio Mínimo (<i>Minimum Mean Square Error</i>)
MSA	Autoatenção de Múltiplas Cabeças (<i>Multi-Head Self-Attention</i>)
MTL	Aprendizado Multitarefa (<i>Multi-Task Learning</i>)
NCC	Correlação Cruzada Normalizada (<i>Normalized Cross-Correlation</i>)
PINN	Rede Neural Informada por Física (<i>Physics-Informed Neural Network</i>)
POE	Órbita Precisa Efeméride (<i>Precise Orbit Ephemerides</i>)
PSNR	Relação Sinal-Ruído de Pico (<i>Peak Signal-to-Noise Ratio</i>)
ReLU	Unidade Linear Retificada (<i>Rectified Linear Unit</i>)
SAR	Radar de Abertura Sintética (<i>Synthetic Aperture Radar</i>)

SNAP	Plataforma de Aplicação Sentinel (<i>Sentinel Application Platform</i>)
SNR	Relação Sinal-Ruído (<i>Signal-to-Noise Ratio</i>)
SQLite	<i>Structured Query Lite</i>
SSIM	Índice de Similaridade Estrutural (<i>Structural Similarity Index</i>)
SSM	Modelo de Espaço de Estado (<i>State Space Model</i>)
SW-MSA	MSA em Janelas Deslocadas (<i>Shifted Window MSA</i>)
TIFF	<i>Tagged Image File Format</i>
TPE	Estimador de Parzen Estruturado em Árvore (<i>Tree-structured Parzen Estimator</i>)
VH	Polarização Vertical-Horizontal
ViT	<i>Vision Transformer</i>
VV	Polarização Vertical-Vertical
W-MSA	MSA em Janelas (<i>Windowed MSA</i>)
XML	Linguagem de Marcação Extensível (<i>Extensible Markup Language</i>)

LISTA DE SÍMBOLOS

A, B, C e D	Matrizes de parâmetros da dinâmica do sistema
C	Número de canais de um tensor de imagem
C_l	Coefficiente de variação local, definido por $\sigma_l / \bar{I}$
$D_{s,i}^k$	Métrica de dissimilaridade entre a imagem i e a imagem de referência k
$F_{alvo}, F_{\hat{I}}$	Função de distribuição acumulada na imagem alvo e prevista, respectivamente
$G_{alvo}, \hat{G}$	Mapa de gradiente para a imagem alvo e restaurada
G_x, G_y	Primeira derivada da imagem na direção horizontal e vertical, respectivamente
g	Ganho multiplicativo aleatório
g_{vv}, g_{vh}	Fator multiplicativo de perturbação na polarização
H	Entropia de Shannon
H_{gumbel}	Histograma da distribuição de Gumbel
H_{pred}	Histograma normalizado do ruído residual
$h(t)$	Vetor de estado latente em um Modelo de Espaço de Estados
I	Intensidade observada do pixel
$\bar{I}$	Intensidade média local em uma janela de vizinhança
$\hat{I}$	Intensidade na imagem prevista
I_h, I_v	Respectivamente: Inversão horizontal e vertical de uma imagem

I_{alvo}	Imagem de referência “limpa”
$I_{ruidosa}$	Imagem de entrada com ruído speckle
$J_{i,j}$	Intensidade final de cada pixel
J_k	Intensidade filtrada para a imagem k
J_s^k	Intensidade final de um pixel s na imagem de referência k
k	Fator de amortecimento (<i>damping factor</i>) no filtro de Frost
L	Número de visadas
l, c e s	Respectivamente: luminância, contraste e estrutura
M	Número de imagens na série temporal
N	Número total de pixels
$M_G(i, j)$	Máscara multiplicativa gaussiana 2D
n_s	Ruído sintético amostrado de uma distribuição Gama
$P(i), Q(i)$	Distribuições de probabilidade discreta do alvo e imagem restaurada
$p(i)$	Probabilidade de ocorrência do i -ésimo nível de cinza em um histograma
R	Refletividade do solo
$\hat{R}$	Estimativa do valor do pixel limpo (R) após a aplicação de um filtro
S	Ruído Speckle
$\vec{v}(i, j)$	Campo de deslocamento vetorial na imagem
W	Peso adaptativo
$W \times H$	Largura (<i>Width</i>) e Altura (<i>Height</i>) de uma imagem ou <i>patch</i> em pixels.
$x(t), y(t)$	Sequência de entrada e saída, respectivamente, de um estado latente
$\mathbb{E}$	Valor Esperado de uma variável aleatória
$\mathcal{F}$	Transformada Rápida de Fourier
$\mathcal{L}_i$	Função de perda
$\mathcal{N}(0, \sigma^2)$	Distribuição Gaussiana de média zero
α	Parâmetro de ponderação, como no filtro Gamma MAP ou FOM
α_i	Peso da influência de uma imagem no Filtro Stroobants
Γ	Distribuição Gama
$\Delta x, \Delta y$	Deslocamento horizontal e vertical de uma imagem, respectivamente
$\Delta \theta$	Variação no ângulo de incidência local θ
ϵ	Constante próxima a 0 (e.g., 10e-6) para diferenciabilidade
η_{log}	Ruído residual no domínio logarítmico
λ_i	Peso de contribuição de uma função de perda

μ	Média de uma região em uma imagem
μ_i	Média local de uma imagem i na série temporal
μ_k	Média global em uma imagem de referência k
μ_η	Média do speckle no domínio logarítmico
σ	Desvio padrão de uma região em uma imagem
σ_I	Desvio padrão de uma única imagem
$\sigma_{\bar{I}}$	Desvio padrão da média de N imagens
σ_S	Desvio padrão do ruído speckle
σ_I^2	Variância local
$\sigma_{local}^2(i, j)$	Variância local para uma janela local
σ_S^2	Variância estimada do ruído
σ_η^2	Variância do speckle no domínio logarítmico
Ω	Janela local
MAX_I	Valor máximo possível para um pixel na imagem
$\sup_x$	Supremo de um conjunto em relação à variável x

SUMÁRIO

1	INTRODUÇÃO	16
1.1	CONTEXTUALIZAÇÃO	16
1.2	O PROBLEMA DO RUÍDO SPECKLE	16
1.3	JUSTIFICATIVA	16
1.4	PROPOSTA.....	17
1.5	OBJETIVOS	17
1.6	ESTRUTURA DO TRABALHO	18
2	FUNDAMENTAÇÃO TEÓRICA.....	19
2.1	SENSORIAMENTO REMOTO SAR.....	19
2.1.1	Princípios de Funcionamento do SAR.....	19
2.1.2	A Missão Sentinel-1 e seus Produtos de Dados (GRD).....	20
2.1.3	O Fenômeno do Ruído Speckle: Origem e Modelagem Estatística.....	20
2.1.4	Métodos Clássicos de Supressão de Ruído Speckle	21
2.1.4.1	Filtragem Espacial	22
2.1.4.1.1	<i>Filtro Lee.....</i>	22
2.1.4.1.2	<i>Filtro Gamma MAP.....</i>	22
2.1.4.1.3	<i>Filtro Frost.....</i>	23
2.1.4.2	Filtragem Multitemporal.....	23
2.1.4.2.1	<i>Filtro de Média Simples.....</i>	24
2.1.4.2.2	<i>Filtro Stroobants.....</i>	24
2.1.4.2.3	<i>Filtro Quegan Simplificado.....</i>	25
2.2	DEEP LEARNING PARA RESTAURAÇÃO DE IMAGENS	25
2.2.1	A Arquitetura U-Net.....	26
2.2.2	Backbones para Modelagem de Contexto Global.....	28
2.2.2.1	O Swin Transformer e a Autoatenção em Janelas	28
2.2.2.2	Modelos de Espaço de Estados: A Arquitetura Mamba	30
2.2.3	Estratégias de Treinamento e Funções de Perda	31
2.2.3.1	Treinamento Supervisionado: Otimização Perceptual.....	31
2.2.3.1.1	<i>Função de Perda Charbonnier.....</i>	32
2.2.3.1.2	<i>Função de Perda SSIM</i>	32
2.2.3.1.3	<i>Função de Perda de Gradiente.....</i>	33
2.2.3.1.4	<i>Função de Perda de Frequência.....</i>	33

2.2.3.2	Aprendizado Multitarefa.....	34
2.2.3.3	Refinamento Não Supervisionado Informado por Física (PINN)	34
2.2.3.3.1	<i>Função de Perda de Variância.....</i>	36
2.2.3.3.2	<i>Função de Perda de Média do Speckle.....</i>	36
2.2.3.3.3	<i>Função de Perda de Histograma</i>	36
2.3	AUMENTO DE DADOS (DATA AUGMENTATION)	37
2.3.1	Augmentações Perceptuais	37
2.3.1.1	Inversão Horizontal e Vertical.....	38
2.3.1.2	Rotações Discretas.....	38
2.3.2	Augmentações Físicas.....	38
2.3.2.1	Adição de Speckle Sintético	38
2.3.2.2	Simulação de Deslocamento Orbital	39
2.3.2.3	Adição de Ruído Térmico.....	39
2.3.2.4	Erro de Calibração Radiométrica.....	40
2.3.2.5	Simulação de Atenuação Atmosférica.....	40
2.3.2.6	Simulação de Degradação Espacial	40
2.3.2.7	Variação do Ângulo de Incidência	41
2.3.2.8	Perturbação da Resposta de Polarização.....	41
2.3.2.9	Distorção Geométrica Não-Linear.....	41
2.3.2.10	Mecanismo de Aplicação Probabilístico	42
2.4	MÉTRICAS PARA AVALIAÇÃO DE QUALIDADE DE IMAGEM.....	43
2.4.1	Métricas Referenciais	44
2.4.1.1	Peak Signal-to-Noise Ratio (PSNR).....	44
2.4.1.2	Structural Similarity Index (SSIM)	45
2.4.1.3	Edge Preservation Degree (EPD)	45
2.4.1.4	Pratt's Figure of Merit (FOM)	45
2.4.1.5	Distância de Kolmogorov-Smirnov (KS)	46
2.4.1.6	Divergência de Kullback-Leibler (KL)	46
2.4.1.7	Entropia de Shannon.....	46
2.4.2	Métricas Não-Referenciais.....	47
2.4.2.1	Equivalent Number of Looks (ENL)	47
2.4.2.2	Edge Preservation Index (EPI)	48
2.4.2.3	Razão das Médias (Ratio of Means).....	48
2.4.2.4	Speckle Index.....	48

2.4.2.5	Matriz de Coocorrência de Níveis de Cinza (GLCM).....	49
2.4.2.6	Entropia Local (Local Entropy).....	49
3	MATERIAIS E MÉTODOS	50
3.1	ESTRUTURA DO FRAMEWORK.....	50
3.2	PIPELINE DE PRÉ-PROCESSAMENTO DE DADOS	53
3.2.1	Área de Estudo e Conjunto de Dados Sentinel-1	53
3.2.2	Correção Radiométrica e Geométrica com ESA SNAP	55
3.2.3	Corregistro da Série Temporal.....	56
3.2.4	Geração de Alvos Multitemporais.....	58
3.2.5	Cálculo de Estatísticas Globais do Conjunto de Dados.....	62
3.3	O DATALOADER CUSTOMIZADO	64
3.3.1	Amostragem Eficiente em Imagens SAR.....	64
3.3.2	Implementação da Amostragem e Cache por Localização	65
3.3.3	Pipeline de Augmentação de Dados Implementado	67
3.4	ARQUITETURA DO MODELO MASF IMPLEMENTADA	68
3.4.1	Estrutura Base U-Net e os Módulos Convolucionais.....	68
3.4.2	Implementação do Bottleneck de Atenção	69
3.4.3	Fusão Inteligente de Características com SkipGate	70
3.4.4	Implementação da Arquitetura Multitarefa	72
3.5	ESTRATÉGIA DE TREINAMENTO E OTIMIZAÇÃO.....	74
3.5.1	Implementação do Treinamento Supervisionado	74
3.5.2	Implementação do Refinamento Não Supervisionado	75
3.5.3	Otimização de Hiperparâmetros com Optuna.....	76
3.6	PROCEDIMENTO EXPERIMENTAL	77
3.6.1	Configuração do Ambiente Computacional.....	78
3.6.2	Métricas de Avaliação Adotadas	79
3.6.3	Métodos de Referência (Baselines) para Comparação.....	80
3.6.4	Estrutura dos Estudos de Ablação	81
4	RESULTADOS E DISCUSSÃO.....	83
4.1	OTIMIZAÇÃO DE HIPERPARÂMETROS E SELEÇÃO DO MODELO	83
4.2	ANÁLISE QUANTITATIVA COMPARATIVA.....	86
4.3	ANÁLISE QUALITATIVA COMPARATIVA	90
4.4	VISUALIZAÇÃO E INTERPRETAÇÃO DOS MAPAS DE ATENÇÃO	93
4.4.1	Análise Qualitativa dos Mapas de Atenção	95

4.4.2	Análise da Correlação entre Atenção e Textura da Imagem.....	96
4.5	DISCUSSÃO DOS ESTUDOS DE ABLAÇÃO	97
5	CONCLUSÃO.....	100
5.1	SÍNTESE DOS RESULTADOS	100
5.2	CONTRIBUIÇÕES DO TRABALHO.....	101
5.3	LIMITAÇÕES E TRABALHOS FUTUROS	101
	REFERÊNCIAS.....	103
	GLOSSÁRIO	107
	APÊNDICE A – PRODUTOS DO SENTINEL-1 UTILIZADOS.....	115
	APÊNDICE B – PARÂMETROS DO MODELO	117
	APÊNDICE C – REPOSITÓRIO DO CÓDIGO-FONTE DO MASF	124

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

O Sensoriamento Remoto por Radar de Abertura Sintética (SAR) é uma tecnologia fundamental para o monitoramento contínuo da superfície terrestre. Sistemas orbitais como o Sentinel-1, da Agência Espacial Europeia (ESA), fornecem um fluxo constante de imagens de alta resolução, independentemente das condições de iluminação solar ou da cobertura de nuvens, permitindo uma vasta gama de aplicações que vão desde a gestão de desastres naturais e agricultura de precisão até o monitoramento de infraestruturas e mapeamento do uso e cobertura do solo. A capacidade de adquirir dados de forma consistente e sistemática faz do SAR uma tecnologia indispensável para a compreensão da dinâmica ambiental em escala global.

1.2 O PROBLEMA DO RUÍDO SPECKLE

Apesar de suas vantagens, a qualidade das imagens SAR é intrinsecamente limitada por um fenômeno inerente ao processo de aquisição coerente: o ruído speckle. Manifestando-se como uma textura granular de alta frequência, o speckle não é um ruído aditivo tradicional, mas sim um ruído multiplicativo, cuja variância é proporcional à intensidade do sinal retroespalhado (LEE, 1986). Essa característica degrada a qualidade radiométrica da imagem, dificulta a interpretação visual e compromete o desempenho de algoritmos de segmentação, classificação e detecção de mudanças, tornando a sua supressão uma etapa crítica no pré-processamento de dados SAR.

1.3 JUSTIFICATIVA

Tradicionalmente, a mitigação do speckle tem sido abordada por meio de filtros espaciais, como os filtros de Lee, Lee Sigma, Gamma MAP e Frost, que operam com base em estatísticas locais da imagem (QUEGAN; YU, 2001). Embora eficazes na suavização de áreas homogêneas, esses métodos frequentemente introduzem um efeito de borramento que resulta na perda de detalhes texturais finos e na degradação de bordas e estruturas lineares, que são cruciais para a análise das imagens. Com o advento do *Deep Learning*, novas abordagens baseadas em Redes Neurais Convolucionais (CNNs), especialmente arquiteturas codificador-

decodificador como a U-Net (RONNEBERGER; FISCHER; BROX, 2015), demonstraram-se superiores para tarefas de restauração de imagem quando comparadas a algoritmos tradicionais. Contudo, a eficácia desses modelos depende da disponibilidade de grandes volumes de dados de treinamento (pares de imagens com e sem ruído) e da capacidade de aprender representações que capturem o contexto espacial e as propriedades estatísticas do ruído.

1.4 PROPOSTA

Para superar as limitações das abordagens existentes, este trabalho propõe o MASF (*Multitemporal Anti-Speckle Framework*), um *framework* para a supressão de ruído speckle em imagens Sentinel-1. As principais inovações do MASF são sua metodologia híbrida e arquitetura moderna. O *framework* emprega uma estratégia de treinamento em dois estágios: um primeiro estágio supervisionado, que utiliza alvos "limpos" (com ruído reduzido) gerados a partir de filtragem multitemporal, e um segundo estágio de refinamento não supervisionado e informado por física (RAISSI; PERDIKARIS; KARNIADAKIS, 2019), onde o modelo é ajustado para respeitar as propriedades estatísticas conhecidas do speckle, sem a necessidade de uma imagem de referência. Em sua arquitetura, o MASF integra mecanismos de atenção avançados, como o *Swin Transformer* (LIU *et al.*, 2021) ou o *Mamba* (GU; DAO, 2023), no *bottleneck* (gargalo) da U-Net para capturar dependências espaciais de longo alcance de forma eficiente.

1.5 OBJETIVOS

O objetivo deste trabalho é desenvolver e avaliar um *framework* de *Deep Learning* de dois estágios para a supressão de ruído speckle em imagens SAR Sentinel-1, investigando a eficácia de *backbones* baseados em atenção e a contribuição de um estágio de refinamento não supervisionado informado por física.

Para atingir este objetivo, foram definidos os seguintes marcos:

- a) Desenvolver um *pipeline* de pré-processamento automatizado para dados brutos do Sentinel-1 utilizando a ferramenta ESA SNAP;
- b) Implementar um método robusto para a geração de imagens-alvo "limpas" através da aplicação de filtros multitemporais;
- c) Implementar e comparar dois mecanismos de atenção (*Swin Transformer* e *Mamba*) no *bottleneck* da U-Net;

- d) Desenvolver e aplicar a estratégia de treinamento em dois estágios (supervisionado e de refinamento informado por física);
- e) Realizar avaliações quantitativas e qualitativas do *framework*, comparando seu desempenho com filtros de speckle clássicos;
- f) Analisar a interpretabilidade dos modelos através da visualização dos mapas de atenção internos.

1.6 ESTRUTURA DO TRABALHO

Este trabalho está estruturado em cinco capítulos. O Capítulo 1 apresenta a contextualização do problema, a justificativa e os objetivos da pesquisa. O Capítulo 2 aborda a fundamentação teórica sobre Sensoriamento Remoto por Radar, o ruído speckle e as tecnologias de *Deep Learning* aplicadas. O Capítulo 3 detalha os materiais e a metodologia completa do MASF, desde o pré-processamento dos dados até a aplicação experimental. O Capítulo 4 apresenta e discute os resultados obtidos, incluindo as análises quantitativas, qualitativas e de interpretabilidade. Por fim, o Capítulo 5 apresenta as conclusões do trabalho, suas contribuições, limitações e aponta direções para pesquisas futuras.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta a fundamentação teórica na qual se baseia o MASF. A seção 2.1 é uma revisão dos princípios do Sensoriamento Remoto por SAR, com ênfase na missão Sentinel-1 e na caracterização do ruído speckle. A seção 2.2 aborda as tecnologias de *Deep Learning* que fundamentam a solução proposta, incluindo as arquiteturas de rede, os mecanismos de atenção e os paradigmas de aprendizado aplicados. A seção 2.3 aborda os métodos de aumento e diversificação dos dados de entrada da rede. Por fim, a seção 2.4 aborda as métricas de avaliação quantitativa que serão empregadas.

2.1 SENSORIAMENTO REMOTO SAR

O SAR é uma modalidade de sensoriamento remoto ativo que opera na faixa de micro-ondas do espectro eletromagnético. Diferentemente de sensores ópticos, que dependem da iluminação solar, os sistemas SAR emitem seus próprios pulsos de energia e registram o sinal que é retroespalhado pela superfície terrestre. Essa capacidade de autoiluminação confere aos sistemas SAR a vantagem de operar continuamente e em quaisquer condições atmosféricas, uma vez que as micro-ondas são capazes de penetrar nuvens, névoa e chuva leve, garantindo uma aquisição de dados consistente e confiável (MOREIRA *et al.*, 2013).

2.1.1 Princípios de Funcionamento do SAR

Um sistema SAR mede duas propriedades fundamentais do sinal de radar retroespalhado: sua amplitude e sua fase. A amplitude está relacionada às propriedades dielétricas e à rugosidade da superfície, determinando o brilho de um alvo na imagem. A fase está relacionada à distância entre o sensor e o alvo, sendo fundamental para técnicas interferométricas (InSAR).

A resolução espacial de um sistema de radar é determinada pelo comprimento de sua antena. Para alcançar a alta resolução necessária para aplicações de observação da Terra a partir de uma plataforma orbital, uma antena física de vários quilômetros seria necessária, o que é impraticável. A técnica de Abertura Sintética resolve este problema ao utilizar o movimento da própria plataforma (satélite ou avião) para simular uma antena muito longa. Ao longo de sua trajetória, o sensor emite uma sequência de pulsos e coleta os sinais de retorno de um mesmo alvo a partir de múltiplas posições, combinando-os de forma coerente para "sintetizar" uma

grande abertura de antena e, assim, alcançar uma alta resolução na direção azimutal (CUMMING; WONG, 2005).

A polarização do sinal de micro-ondas descreve a orientação do campo elétrico da onda emitida e recebida. As configurações mais comuns são a copolarizada, onde a polarização de transmissão e recepção é a mesma (*e.g.*, VV - vertical-vertical, HH - horizontal-horizontal), e a de polarização cruzada, onde elas são ortogonais (*e.g.*, VH - vertical-horizontal, HV - horizontal-vertical). Diferentes polarizações interagem de maneiras distintas com a superfície, fornecendo informações complementares sobre a estrutura e a composição dos alvos.

2.1.2 A Missão Sentinel-1 e seus Produtos de Dados (GRD)

A missão Sentinel-1, parte do programa Copernicus da ESA, consiste em uma constelação de satélites (Sentinel-1A, Sentinel-1B e Sentinel-1C) equipados com um sensor SAR operando em Banda-C. Seu principal modo de aquisição sobre áreas terrestres é o *Interferometric Wide Swath* (IW), que oferece uma ampla faixa de imageamento (250 km) com uma resolução espacial moderada (aproximadamente 20 metros) e polarizações duplas (VV+VH ou HH+HV).

Os dados são disponibilizados em diferentes níveis de processamento. Para a maioria das aplicações de análise de imagem, o produto mais utilizado é o *Ground Range Detected* (GRD). Neste nível, os dados já foram multivisados (*multilooked*) para reduzir o ruído speckle inicial, projetados da geometria de aquisição inclinada (*slant range*) para a geometria de solo (*ground range*) e possuem uma amostragem de pixels quadrados. No entanto, os produtos GRD ainda contêm distorções geométricas e radiométricas significativas, além do ruído speckle residual, exigindo um pré-processamento rigoroso antes de sua utilização (FILIPPONI, 2019).

2.1.3 O Fenômeno do Ruído Speckle: Origem e Modelagem Estatística

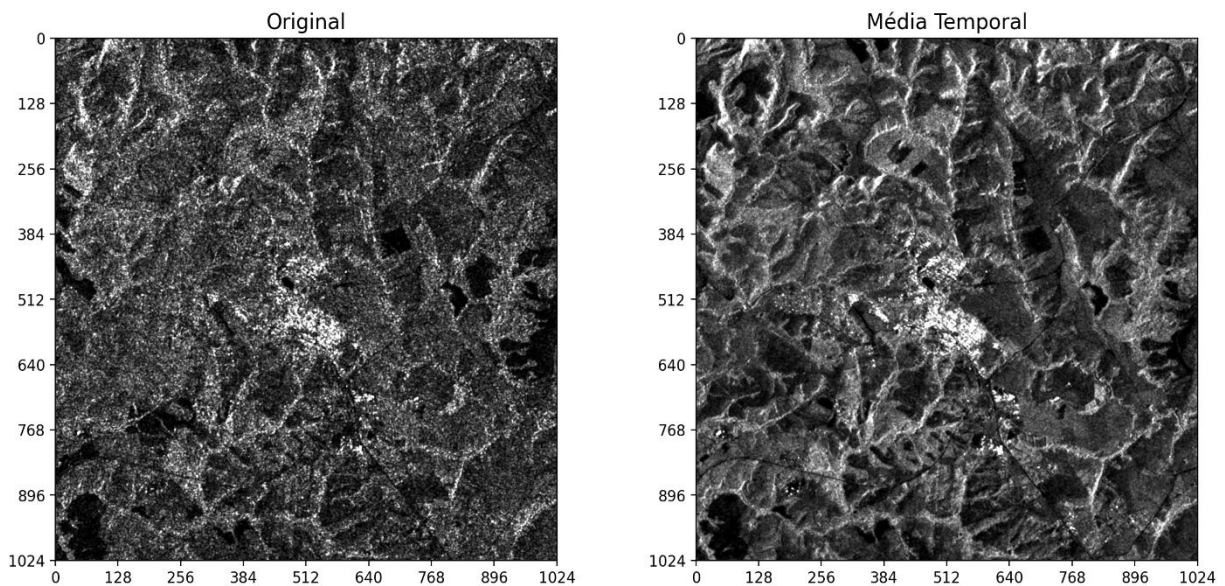
O ruído speckle é uma característica distintiva e desafiadora das imagens SAR. Ele não é um ruído instrumental, mas sim o resultado do processo de imageamento coerente. Dentro de uma única célula de resolução da imagem, existem múltiplos dispersores elementares no solo. O sinal de radar retroespalhado por cada um deles viaja de volta ao sensor com uma fase ligeiramente diferente. A soma vetorial coerente de todos esses sinais no detector do sensor resulta em interferência, que pode ser construtiva (criando um pixel muito brilhante) ou destrutiva (criando um pixel muito escuro).

Esse padrão de interferência aleatória manifesta-se na imagem como uma textura granular, como ilustrado na Figura 1. Matematicamente, o speckle é modelado como um ruído multiplicativo, onde a intensidade observada (I) em um pixel é o produto da refletividade real do solo (R) e o ruído speckle (S):

$$I = R \cdot S \quad (1)$$

A principal consequência deste modelo é que a variância do ruído não é constante, mas sim proporcional à intensidade do sinal. Áreas mais brilhantes na imagem (alto retroespalhamento) exibem uma variação de ruído maior do que áreas escuras. A distribuição estatística da intensidade do speckle é frequentemente modelada por uma distribuição Gama, o que torna sua remoção por métodos lineares particularmente ineficaz (LEE, 1986; OLIVER; QUEGAN, 2004).

Figura 1 – Comparação entre a imagem SAR com speckle e sua média multitemporal



Fonte: Autoria Própria.

2.1.4 Métodos Clássicos de Supressão de Ruído Speckle

A supressão do ruído speckle é um campo de pesquisa ativo há décadas. Os métodos clássicos podem ser categorizados em dois grupos: filtros espaciais e filtros multitemporais.

2.1.4.1 Filtragem Espacial

Os filtros espaciais operam em uma única imagem. Filtros como Lee, Lee Sigma, Gamma MAP e Frost utilizam uma janela deslizante para calcular estatísticas locais e estimar o valor do pixel central com base em seus vizinhos (QUEGAN; YU, 2001). Embora reduzam o ruído, frequentemente esta redução causa o borramento de detalhes finos e bordas.

2.1.4.1.1 Filtro Lee

O Filtro Lee é baseado na minimização do erro quadrático médio (MMSE) sob a premissa de um modelo de ruído multiplicativo. A estimativa do pixel livre de ruído ($\hat{R}$) é uma média ponderada entre a intensidade observada (I) e a média local da vizinhança ($\bar{I}$):

$$\hat{R} = \bar{I} + W \cdot (I - \bar{I}) \quad (2)$$

O peso adaptativo (W) ajusta a intensidade da suavização com base na heterogeneidade local. Ele é calculado a partir da variância local (σ_I^2) e da variância estimada do ruído (σ_S^2):

$$W = \frac{\sigma_I^2}{\sigma_S^2 + \sigma_I^2} \quad (3)$$

Em áreas homogêneas, $\sigma_I^2 \approx \sigma_S^2$, fazendo com que $W \approx 0.5$, resultando em forte suavização. Em áreas suaves, σ_I^2 é grande, $W \approx 1$ e $\hat{R} \approx I$, preservando os detalhes e as bordas da imagem. Uma variação comum, o Filtro Lee Sigma, refina essa abordagem ao incluir na média local apenas os pixels vizinhos que estão dentro de um intervalo de desvio padrão (σ) em torno do pixel central, melhorando a preservação de bordas (LEE, 1983).

2.1.4.1.2 Filtro Gamma MAP

O Filtro Gamma MAP (LOPES; TOUZI; NEZRY, 1990) é baseado em um modelo estatístico mais sofisticado. Ele assume que a distribuição de intensidade do speckle segue uma distribuição Gama e utiliza uma abordagem Bayesiana de Estimação a Posteriori Máxima (MAP) para encontrar o valor de pixel mais provável ($\hat{R}$), dado o valor observado (I). A solução para ($\hat{R}$) é a raiz positiva da seguinte equação quadrática:

$$\alpha \cdot \hat{R}^2 + (\alpha - L - 1) \cdot \bar{I} \cdot \hat{R} - \alpha \cdot L \cdot I \cdot \bar{I} = 0 \quad (4)$$

Onde α é um parâmetro de textura relacionado ao coeficiente de variação:

$$\alpha = \frac{1 + L}{C_i^2 - C_s^2} \quad (5)$$

2.1.4.1.3 Filtro Frost

O Filtro Frost (Frost *et al.*, 1982) é um filtro adaptativo que aplica um *kernel* de ponderação que decai exponencialmente com a distância do pixel central. A taxa de decaimento é modulada pelo coeficiente de variação local (C_I), que é a razão entre o desvio padrão local e a média local. O peso $m(d)$ para um pixel a uma distância d do centro é dado por:

$$m(d) = e^{-k \cdot C_I^2 \cdot d} \quad (6)$$

Onde k é um fator de amortecimento. Em áreas homogêneas, C_I é baixo, resultando em um *kernel* de média simples. Em áreas heterogêneas (bordas), C_I é alto, fazendo com que os pesos decaiam rapidamente, preservando assim a estrutura local.

2.1.4.2 Filtragem Multitemporal

A filtragem multitemporal é uma das abordagens mais eficazes para a redução do speckle, servindo como base para a geração de imagens de referência de alta qualidade neste trabalho. O princípio fundamental reside na estatística do ruído. Assumindo que o retroespalhamento do solo (R) é estático ou varia lentamente ao longo de uma série temporal de (M) imagens corregradas, o padrão de speckle (S) em cada aquisição pode ser considerado uma variável aleatória independente. Ao realizar uma média das intensidades pixel a pixel ao longo do tempo, o valor médio do sinal (R) é preservado, enquanto o ruído é atenuado.

A redução do desvio padrão do ruído speckle (σ_S) é proporcional à raiz quadrada do número de imagens (N) utilizadas na média. Se (σ_I) é o desvio padrão de uma única imagem, o desvio padrão da imagem resultante da média de (M) imagens, (σ_I), é dado por:

$$\sigma_I = \frac{\sigma_I}{\sqrt{M}} \quad (7)$$

Embora este princípio demonstre que, ao aumentar o número de imagens na série temporal, é possível reduzir o speckle a níveis arbitrariamente baixos, produzindo um alvo "limpo" que serve como *ground truth* para o treinamento supervisionado de modelos de *Deep Learning*, a eficácia do método exibe retornos decrescentes. A redução do ruído speckle, proporcional ao inverso da raiz do número de imagens, torna-se marginal para séries temporais muito longas, enquanto o custo computacional associado ao corregristo e processamento continua a escalar linearmente.

A limitação do método de filtragem multitemporal se dá, portanto, no equilíbrio entre o ganho marginal de qualidade e o custo computacional do processo, além do risco da perda de coerência temporal inerente à dinâmica do meio físico. Tal fato estabelece a necessidade de abordagens complementares, como o refinamento orientado pelas propriedades físicas conhecidas do speckle, para otimizar a qualidade da imagem de referência.

Este trabalho implementa três variações de filtros multitemporais, conforme descrito por Smara e Mansoura (2004).

2.1.4.2.1 *Filtro de Média Simples*

É a implementação mais direta do princípio de filtragem temporal. A intensidade final de cada pixel ($J_{i,j}$) é a média aritmética simples das intensidades nas M imagens:

$$J_{i,j} = \frac{1}{M} \sum_{k=1}^M I_k(i,j) \quad (8)$$

Apesar de sua simplicidade e eficácia na supressão do ruído, este método pode borrar feições que sofrem mudanças radiométricas reais ao longo do tempo.

2.1.4.2.2 *Filtro Stroobants*

Esta abordagem, proposta por Stroobants (HUOT *et al.*, 1998), refina a média simples ao introduzir um sistema de ponderação para torná-la mais robusta a mudanças temporais. A intensidade final (J_s^k) de um pixel s na imagem de referência k é uma média ponderada das intensidades normalizadas pela média local (μ_i) de cada imagem i na série temporal:

$$J_s^k = \mu_k \frac{\sum_{i=1}^M \alpha_i \cdot \frac{I_s^i}{\mu_i}}{\sum_{i=1}^M \alpha_i} \quad (9)$$

O peso (α_i) é inversamente proporcional a uma métrica de dissimilaridade ($D_{s,i}^k$) entre a imagem i e a imagem de referência k . Se a dissimilaridade for alta (indicando uma mudança real), o peso α_i é reduzido, diminuindo a influência daquela imagem na média final e preservando melhor as mudanças temporais.

2.1.4.2.3 Filtro Quegan Simplificado

O filtro proposto por Quegan e Le Toan (1998) é uma técnica mais sofisticada que visa produzir M imagens filtradas a partir da série temporal. Na sua versão simplificada, utilizada neste trabalho para o caso de imagens não correlacionadas (comum em séries temporais com grande intervalo), a intensidade filtrada (J_k) para a imagem k é uma média das intensidades de todas as imagens, normalizadas por suas respectivas médias locais (μ_i):

$$J_k = \frac{\mu_k}{M} \sum_{i=1}^M \frac{I_i}{\mu_i} \quad (10)$$

Este método normaliza a radiometria de cada imagem antes da média, tornando o filtro robusto a variações de calibração ou condições de aquisição, ao mesmo tempo que preserva a resolução espacial.

2.2 DEEP LEARNING PARA RESTAURAÇÃO DE IMAGENS

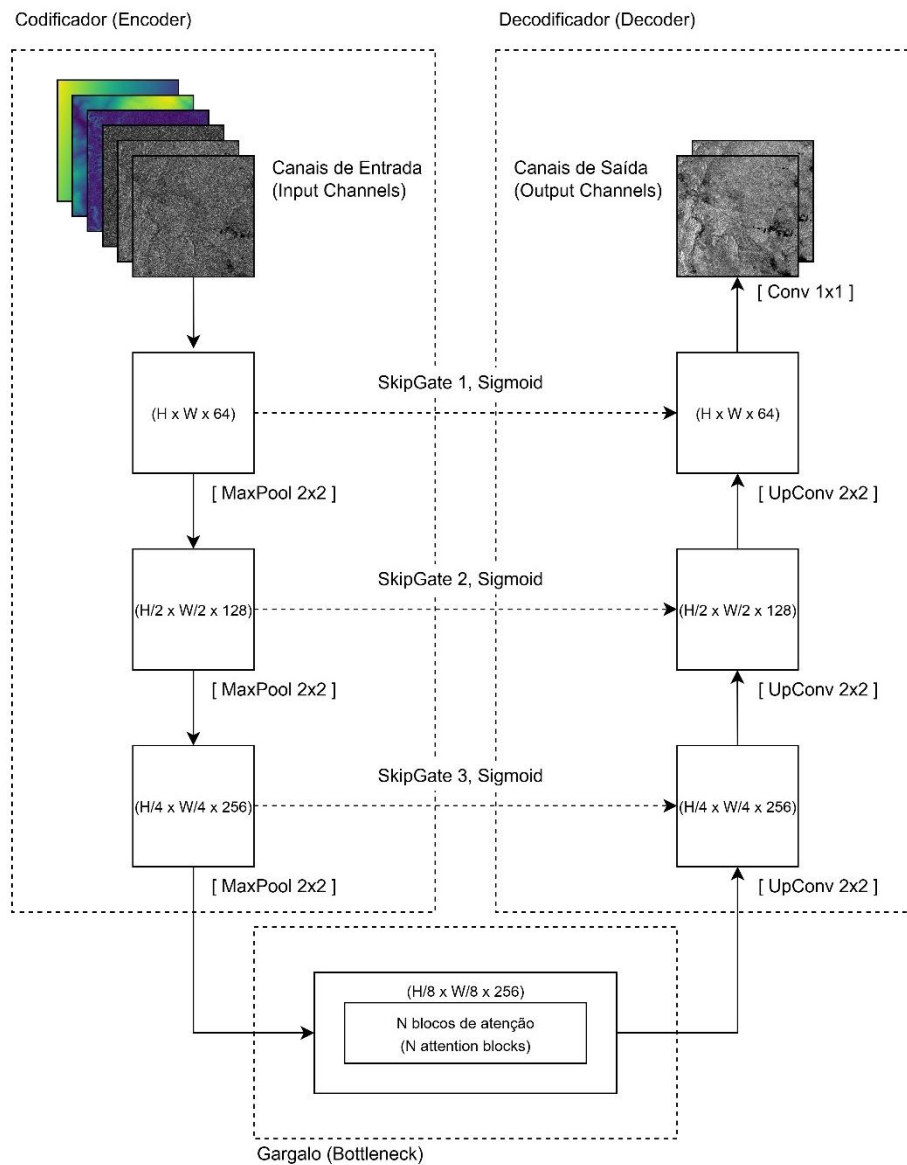
Enquanto os filtros clássicos representam abordagens algorítmicas baseadas em modelos estatísticos predefinidos, o *Deep Learning* se apresenta como um paradigma alternativo, capaz de aprender a representação dos dados para uma tarefa específica. Para o problema da supressão de ruído speckle, as CNNs e arquiteturas mais recentes baseadas em atenção oferecem uma capacidade superior de modelar as complexas relações espaciais e de textura de uma imagem, resultando em uma restauração que preserva a fidelidade dos detalhes de forma mais eficaz que os métodos tradicionais (ZHANG *et al.*, 2017). Esta superioridade é consequência da habilidade das redes profundas em aprender hierarquias de características

diretamente dos dados, desde bordas simples e texturas em camadas iniciais até dependências complexas em camadas mais profundas, um processo que se mostra particularmente eficaz para a tarefa de separar o sinal SAR subjacente do componente de ruído speckle.

2.2.1 A Arquitetura U-Net

A arquitetura U-Net, proposta originalmente por Ronneberger, Fischer e Brox (2015) para segmentação de imagens biomédicas, tornou-se um padrão para uma vasta gama de tarefas de tradução de imagem para imagem, como demonstrado por Isola *et al.* (2017), incluindo a remoção de ruído. Sua estrutura é caracterizada por um caminho de contração (codificador) e um caminho de expansão simétrico (decodificador), que lhe conferem um formato de "U" e um balanço entre a extração de contexto e a localização precisa de características.

Figura 2 – Diagrama da arquitetura U-Net



Fonte: Autoria Própria.

O codificador consiste em uma sequência de blocos convolucionais seguidos por operações de *pooling* (geralmente *max-pooling*). A cada passo, os blocos convolucionais, através de seus *kernels* treináveis, atuam como extratores de características, aprendendo a identificar padrões relevantes na imagem. A operação de *pooling* subsequente reduz a resolução espacial do mapa de características, o que cumpre duas funções essenciais: agregar as informações locais, tornando a representação mais robusta a pequenas translações; aumentar o "campo receptivo" efetivo das camadas subsequentes. Isso significa que um neurônio em uma camada mais profunda "enxerga" uma porção maior da imagem original, permitindo que a rede aprenda informações contextuais em diferentes escalas, desde texturas locais até a estrutura global da cena.

De forma simétrica, o decodificador trabalha para reconstruir a imagem detalhada a partir do mapa de características de baixa resolução e alta semântica gerado pelo codificador. Ele utiliza operações de *upsampling* (como a convolução transposta ou a interpolação bilinear) para aumentar gradualmente a resolução espacial. Cada etapa de *upsampling* é seguida por blocos convolucionais que refinam os detalhes da imagem reconstruída, transformando as informações contextuais abstratas de volta em representações espaciais precisas.

A inovação crucial da U-Net são as conexões de atalho (*skip connections*), que concatenam diretamente os mapas de características do codificador aos mapas de características de mesma resolução no decodificador. Essas conexões são vitais, pois formam um atalho para o fluxo de gradiente durante o treinamento, combatendo o problema do gradiente evanescente que ocorre em redes muito profundas. Além disso, elas permitem que o decodificador recupere informações espaciais de alta frequência (como bordas e texturas finas) que seriam inevitavelmente perdidas durante as operações de *pooling* no codificador. Ao combinar a informação de alta resolução das *skip connections* com a informação semântica do decodificador, a U-Net possibilita uma reconstrução de imagem notavelmente precisa. No MASF, essas conexões são aprimoradas com um mecanismo de atenção, o *SkipGate*, para uma fusão de características ainda mais inteligente e adaptativa.

2.2.2 Backbones para Modelagem de Contexto Global

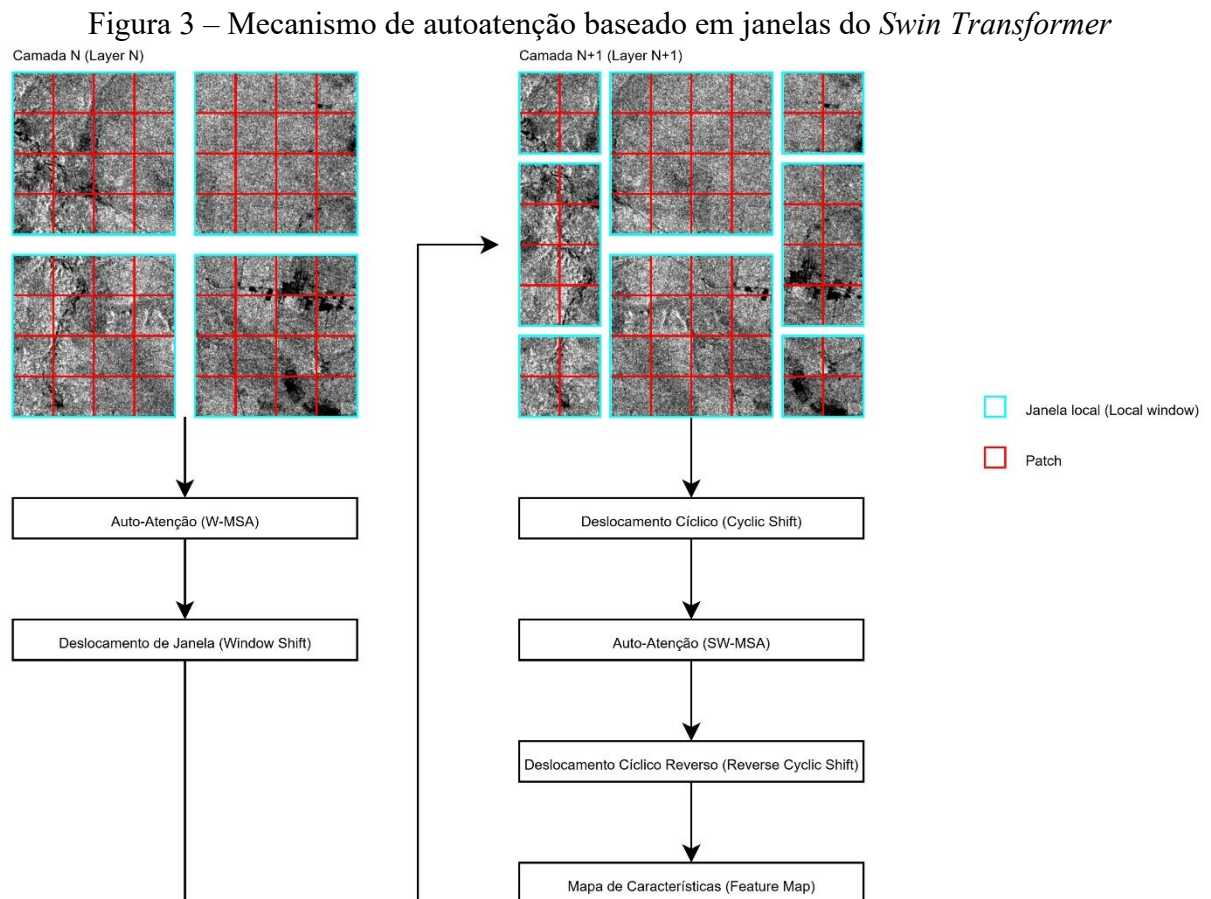
O "gargalo" (*bottleneck*) da U-Net, a camada mais profunda que conecta o codificador ao decodificador, é o local ideal para integrar módulos capazes de modelar dependências de longo alcance e capturar o contexto global da imagem. O MASF explora duas arquiteturas para esta finalidade, superando as limitações do campo receptivo inerentemente local das convoluções tradicionais.

2.2.2.1 O Swin Transformer e a Autoatenção em Janelas

O *Swin Transformer* (LIU *et al.*, 2021) é uma arquitetura de *Vision Transformer* (ViT) que introduziu um mecanismo de atenção hierárquico e computacionalmente eficiente. Diferentemente dos ViTs originais que calculam a autoatenção globalmente, um processo com custo computacional quadrático em relação ao número de pixels, tornando-o inviável para imagens de alta resolução, o *Swin Transformer* adota uma estratégia local-global que opera em duas etapas alternadas:

a) *Windowed Multi-Head Self-Attention* (W-MSA): A imagem é dividida em janelas locais não sobrepostas, e a autoatenção é calculada apenas entre os patches de imagem dentro de cada janela. Isso reduz drasticamente o custo computacional, tornando-o linear em relação ao número de pixels e permitindo a construção de modelos profundos. A limitação, contudo, é que a informação fica confinada dentro de cada janela;

b) *Shifted Window Multi-Head Self-Attention* (SW-MSA): Para permitir a comunicação entre as janelas, na camada seguinte, as janelas são deslocadas em metade de seu comprimento. Essa mudança de partição cria janelas que abrangem patches que pertenciam a janelas diferentes na camada anterior. Ao calcular a autoatenção nessas janelas deslocadas, o modelo estabelece conexões entre regiões vizinhas.



Fonte: Autoria Própria.

Essa alternância entre W-MSA e SW-MSA permite que o modelo construa representações globais de forma eficiente, camada por camada, com o campo receptivo de cada *patch* crescendo exponencialmente com a profundidade da rede. Integrado ao *bottleneck* da U-

Net, o *Swin Transformer* dá ao modelo a capacidade de capturar dependências espaciais de longo alcance, modelando o contexto de forma mais robusta que as convoluções.

2.2.2.2 Modelos de Espaço de Estados: A Arquitetura Mamba

O *Mamba* (GU; DAO, 2023) representa uma alternativa aos modelos baseados em atenção, utilizando a arquitetura de Modelo de Espaço de Estados (SSM). Os SSMs são inspirados na teoria de controle clássica e são projetados para modelar sequências de dados. Eles mapeiam uma sequência de entrada $x(t)$ para uma saída $y(t)$ através de um estado latente implícito $h(t)$ com as equações:

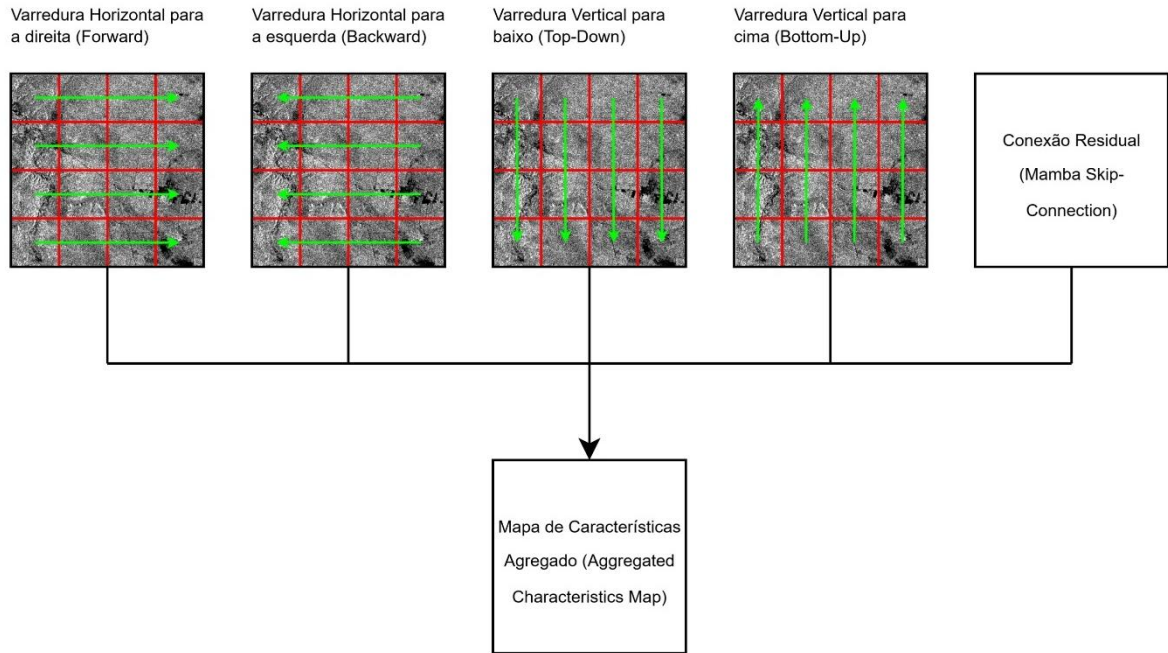
$$h'(t) = A \cdot h(t) + B \cdot x(t) \quad (11)$$

$$y(t) = C \cdot h(t) + D \cdot x(t) \quad (12)$$

Onde A , B , C e D são matrizes de parâmetros da dinâmica do sistema e são aprendidas durante o treinamento. A inovação disruptiva do *Mamba* é seu mecanismo de seleção de estado, que torna essas matrizes dependentes da entrada. Isso confere ao modelo a capacidade de filtrar informações irrelevantes e reter seletivamente as informações contextuais mais importantes ao longo da sequência, resolvendo uma limitação histórica dos SSMs em lidar com dependências de longo prazo.

Para aplicar essa arquitetura inerentemente 1D a dados 2D como imagens, o MASF adota uma abordagem de varredura multidirecional: o mapa de características 2D é achatado em sequências 1D em quatro direções (horizontal da esquerda para a direita, da direita para a esquerda; e vertical de cima para baixo, de baixo para cima), processado por instâncias separadas do *Mamba*, e os resultados são agregados. Essa estratégia permite que o *Mamba* construa uma compreensão holística do contexto espacial com complexidade computacional linear, oferecendo uma alternativa altamente eficiente aos *Transformers*.

Figura 4 – Diagrama da varredura multidirecional para aplicação do *Mamba* em dados 2D



Fonte: Autoria Própria.

2.2.3 Estratégias de Treinamento e Funções de Perda

A escolha da função de perda e da estratégia de treinamento é tão crucial quanto a arquitetura do modelo para alcançar resultados de alta qualidade na restauração de imagens. Este trabalho emprega uma abordagem multifacetada, combinando um estágio de treinamento supervisionado, que utiliza uma perda composta para otimizar a qualidade perceptual, com um estágio de refinamento não supervisionado, que se baseia em perdas informadas por física para garantir a fidelidade estatística do resultado. Além disso, explora-se o paradigma de aprendizado multitarefa como uma forma de regularização indutiva.

2.2.3.1 Treinamento Supervisionado: Otimização Perceptual

O treinamento inicial do MASF é supervisionado, utilizando pares de imagens “ruidosas” ($I_{entrada}$) e seus respectivos alvos multitemporais “limpos” (I_{alvo}). Funções de perda simples, como o erro L1 (diferença absoluta) ou L2 (erro quadrático médio), embora matematicamente convenientes, são notoriamente limitadas em capturar a qualidade perceptual da imagem. Elas penalizam qualquer desvio pixel a pixel de forma uniforme, o que incentiva o modelo a encontrar uma média “segura” de todas as soluções possíveis, frequentemente

resultando em saídas visualmente borradas e com texturas suavizadas, uma limitação bem documentada das métricas de erro por pixel (JOHNSON; ALAHI; FEI-FEI, 2016).

Para superar essa limitação, este trabalho utiliza uma perda composta perceptual ($\mathcal{L}_{perceptual}$), que é uma soma ponderada de múltiplas funções de perda, cada uma focada em um aspecto diferente da qualidade percebida por uma analista humano. A perda total é dada por:

$$\mathcal{L}_{perceptual} = \lambda_1 \cdot \mathcal{L}_{charbonnier} + \lambda_2 \cdot \mathcal{L}_{ssim} + \lambda_3 \cdot \mathcal{L}_{grad} + \lambda_4 \cdot \mathcal{L}_{freq} \quad (13)$$

Onde λ_i são os pesos da contribuição de cada componente.

2.2.3.1.1 Função de Perda Charbonnier

A perda Charbonnier é uma variante suave da perda L1, mais robusta a *outliers* (BARRON, 2017). Ela mede o erro pixel a pixel entre a imagem prevista ($\hat{I}$) e a alvo (I_{alvo}), mas com uma curvatura suave próxima de zero que evita gradientes explosivos. É definida como:

$$\mathcal{L}_{charbonnier} = \frac{1}{N} \sum_{i=1}^N \sqrt{(I_{alvo,i} - \hat{I}_i)^2 + \epsilon^2} \quad (14)$$

Onde N é o número total de pixels e ϵ é uma pequena constante (*e.g.*, $10e-6$) que garante a diferenciabilidade.

2.2.3.1.2 Função de Perda SSIM

A perda SSIM é baseada no Índice de Similaridade Estrutural (WANG *et al.*, 2004), que compara janelas locais de duas imagens com base em três componentes: luminância (l), contraste (c) e estrutura (s). A perda é calculada como a diferença de similaridade, incentivando o modelo a preservar a organização estrutural da cena:

$$\mathcal{L}_{ssim} = 1 - SSIM(I_{alvo}, \hat{I}) \quad (15)$$

O SSIM é projetado para se correlacionar melhor com a percepção humana de qualidade do que as métricas baseadas em erro absoluto.

2.2.3.1.3 *Função de Perda de Gradiente*

A perda de gradiente penaliza as diferenças nos gradientes (bordas) entre a imagem prevista e a alvo, forçando o modelo a gerar resultados mais nítidos. É comumente calculada aplicando-se um operador de Sobel, que aproxima a primeira derivada da imagem nas direções horizontal (G_x) e vertical (G_y), e computando o erro L1 entre os mapas de gradiente resultantes:

$$\mathcal{L}_{grad} = \|G_x(I_{alvo}) - G_x(\hat{I})\|_1 + \|G_y(I_{alvo}) - G_y(\hat{I})\|_1 \quad (16)$$

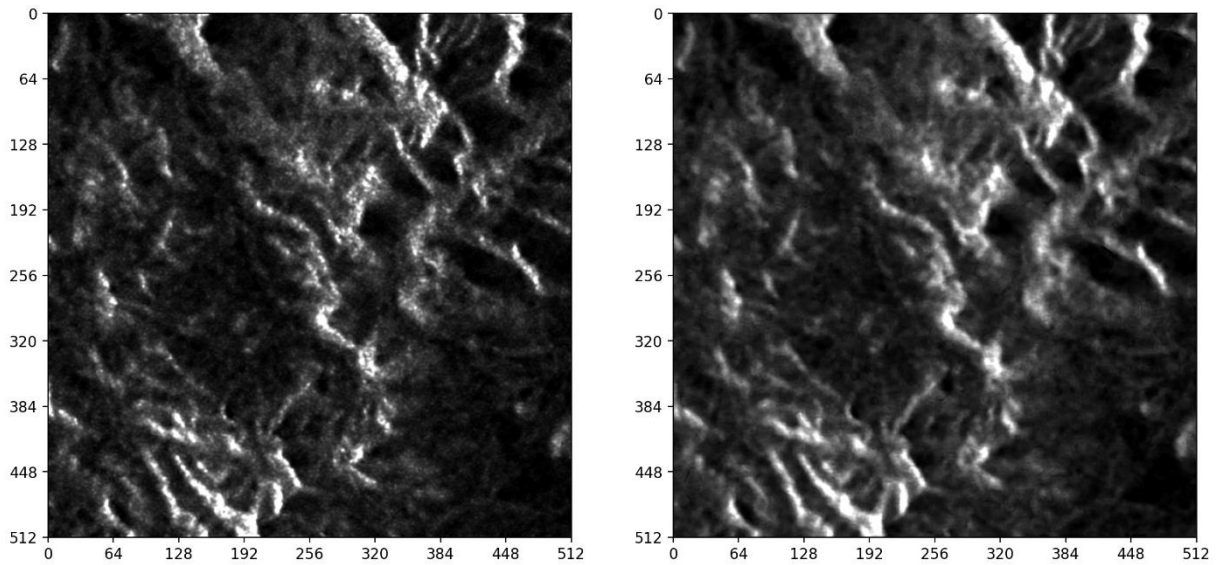
2.2.3.1.4 *Função de Perda de Frequência*

A perda de frequência opera no domínio da frequência, comparando as Transformadas de Fourier das imagens. Ao calcular o erro entre as magnitudes dos espectros de frequência, esta perda força o modelo a reconstruir corretamente não apenas as componentes de baixa frequência (áreas suaves), mas também as de alta frequência (texturas finas e detalhes), que são frequentemente perdidas com perdas puramente espaciais.

$$\mathcal{L}_{freq} = \||\mathcal{F}(I_{alvo})| - |\mathcal{F}(\hat{I})|\|_1 \quad (17)$$

Onde $\mathcal{F}$ denota a Transformada Rápida de Fourier (FFT).

Figura 5 – Imagem restaurada com L1 puro vs. perda perceptual composta



Fonte: Autoria Própria.

Legenda: A figura ilustra um patch restaurado utilizando a função de perda L1 (à direita) e função de perda perceptual composta (à esquerda).

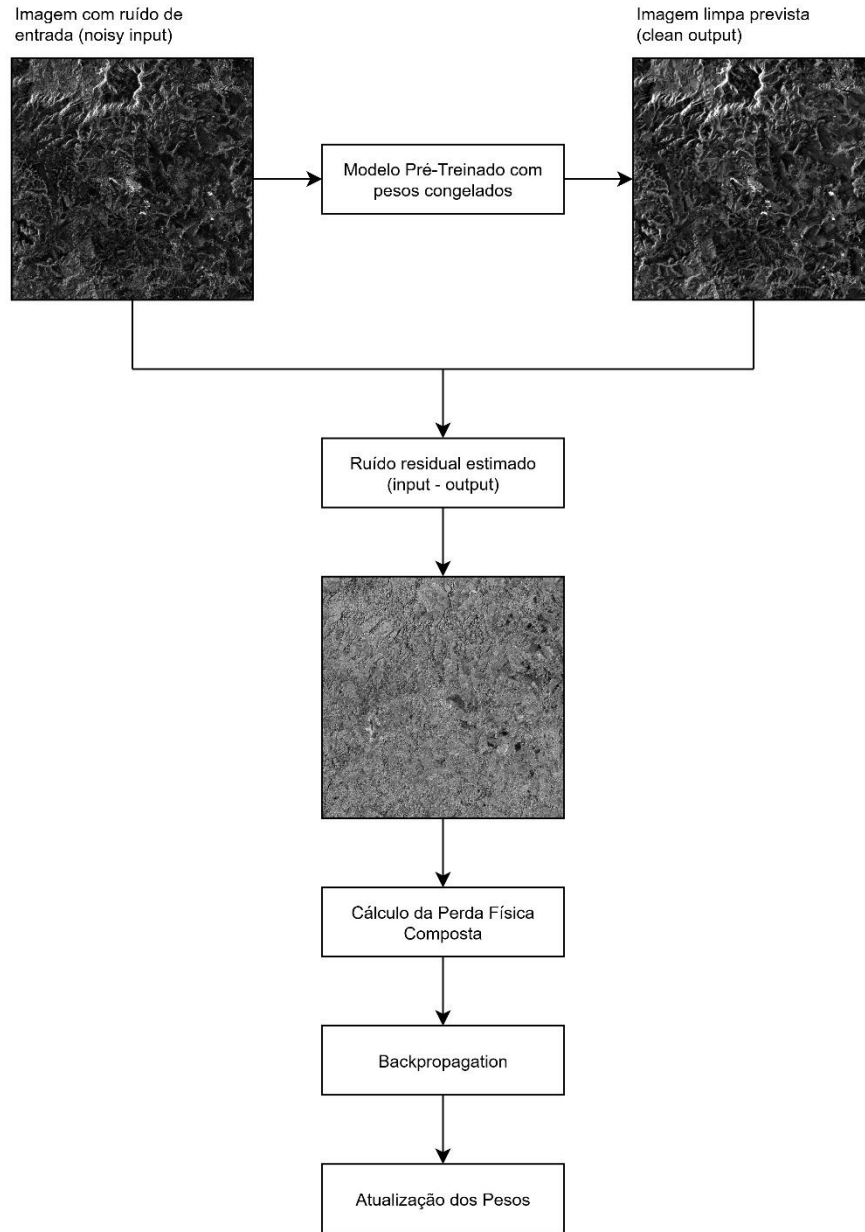
2.2.3.2 Aprendizado Multitarefa

O aprendizado multitarefa é um paradigma onde um modelo é treinado para realizar múltiplas tarefas simultaneamente. No MASF, o modelo pode ser configurado para duas tarefas: a reconstrução da imagem limpa ($\mathcal{L}_{perceptual}$) e a predição da máscara de ruído residual ($\mathcal{L}_{ruído}$), que é supervisionada pela diferença entre a entrada com ruído e o alvo. A hipótese é que, ao forçar o modelo a aprender explicitamente a estrutura estatística do ruído, ele desenvolve uma representação interna mais robusta. Essa tarefa auxiliar atua como uma forma de regularização indutiva, guiando o modelo para uma solução mais generalizável para a tarefa primária.

2.2.3.3 Refinamento Não Supervisionado Informado por Física (PINN)

O conceito de Redes Neurais Informadas por Física (*Physics-Informed Neural Networks* - PINN) envolve a incorporação de leis físicas como um termo de regularização na função de perda (RAISSI; PERDIKARIS; KARNIADAKIS, 2019). O MASF adota este paradigma em um estágio de refinamento opcional e não supervisionado, que aborda a limitação da ausência de alvo multitemporal perfeito no mundo real.

Figura 6 – Diagrama conceitual do refinamento informado por física



Fonte: Autoria Própria.

Após o treinamento supervisionado, o modelo é refinado utilizando apenas imagens com ruído. A função de perda avalia se as propriedades estatísticas do ruído residual no domínio logarítmico (η_{log}) estão de acordo com o modelo teórico do speckle. O ruído residual é definido como a diferença entre a entrada logarítmica e a predição logarítmica do modelo: $\eta_{log} = \log(1 + I_{entrada}) - \log(1 + \hat{I})$. A perda física composta ($\mathcal{L}_{física}$) é então:

$$\mathcal{L}_{física} = \lambda_1 \cdot \mathcal{L}_{variância} + \lambda_2 \cdot \mathcal{L}_{média} + \lambda_3 \cdot \mathcal{L}_{histograma} \quad (18)$$

Este estágio permite que o modelo faça um ajuste fino em sua saída para ser estatisticamente mais realista, melhorando a fidelidade física do resultado, mesmo na ausência de uma referência limpa.

2.2.3.3.1 *Função de Perda de Variância*

A teoria indica que, no domínio logarítmico, o speckle se comporta como um ruído aditivo com variância constante (σ_η^2), independente do sinal. Esta perda penaliza a variação espacial da variância local do ruído residual, forçando-a a ser homogênea. Para uma janela local (Ω), a variância local é $\sigma_{local}^2(i, j) = Var_{p \in \Omega_{i,j}} \cdot \eta_{log}(p)$. A perda é a variância deste mapa de variâncias:

$$\mathcal{L}_{variância} = Var(\sigma_{local}^2) \quad (19)$$

Minimizar esta perda força σ_{local}^2 a se aproximar de uma constante em toda a imagem.

2.2.3.3.2 *Função de Perda de Média do Speckle*

A média teórica do ruído speckle no domínio logarítmico (μ_η) é próxima de zero. Esta perda penaliza desvios da média global do ruído residual em relação a este valor esperado, evitando que o modelo introduza vieses de brilho. É definida como o valor absoluto da média do ruído residual:

$$\mathcal{L}_{média} = |\mathbb{E}[\eta_{log}]| \quad (20)$$

2.2.3.3.3 *Função de Perda de Histograma*

Esta é a restrição física mais forte, pois compara a distribuição de probabilidade empírica do ruído residual com a distribuição teórica de Gumbel, que é uma boa aproximação para o speckle no domínio logarítmico. A perda é calculada como a distância (e.g., L1) entre o histograma normalizado do ruído residual (H_{pred}) e a Função de Densidade de Probabilidade teórica de Gumbel (H_{gumbel}):

$$\mathcal{L}_{\text{histograma}} = \|H_{\text{pred}} - H_{\text{gumbel}}\|_1 \quad (21)$$

2.3 AUMENTO DE DADOS (DATA AUGMENTATION)

Um dos maiores desafios no treinamento de redes neurais profundas, que possuem milhões de parâmetros treináveis, é evitar o sobreajuste (*overfitting*). Este fenômeno ocorre quando o modelo se ajusta excessivamente aos dados de treinamento, memorizando o ruído e as particularidades do conjunto de treino em vez de aprender as relações subjacentes e generalizáveis. Consequentemente, um modelo sobreajustado apresenta um desempenho excelente nos dados de treinamento, mas uma performance medíocre em dados novos e não vistos.

O Aumento de Dados (Augmentação de Dados ou *Data Augmentation*) é uma das técnicas mais eficazes e amplamente utilizadas para mitigar o sobreajuste e melhorar a capacidade de generalização de um modelo. A técnica consiste em expandir artificialmente o conjunto de treinamento através da aplicação de transformações que geram novas amostras sintéticas, porém realistas. Ao expor o modelo a uma variedade maior de dados durante o treinamento, ele é forçado a aprender representações de características que são mais robustas e invariantes a perturbações comuns no domínio do problema. Para a tarefa de restauração de imagens SAR, as estratégias de aumento de dados podem ser divididas em duas categorias principais: perceptuais e físicas.

2.3.1 Augmentações Perceptuais

As augmentações perceptuais consistem em transformações geométricas que não alteram o conteúdo semântico ou a distribuição estatística da imagem. O objetivo é ensinar ao modelo a invariância a certas propriedades, como orientação e simetria, garantindo que a resposta do modelo não mude se um objeto ou textura aparecer em uma orientação diferente. No contexto de um problema de tradução de imagem-para-imagem, como a remoção de ruído, é fundamental que a mesma transformação seja aplicada de forma idêntica tanto à imagem de entrada quanto à sua imagem-alvo correspondente.

2.3.1.1 Inversão Horizontal e Vertical

A inversão de um *patch* de imagem I de dimensões $H \times W$ pode ser representada pela manipulação de seus índices. A inversão horizontal, I_h , e a vertical, I_v , são dadas por:

$$I_h(i, j) = I(i, W - 1 - j) \quad (22)$$

$$I_v(i, j) = I(H - 1 - i, j) \quad (23)$$

Onde (i, j) são os índices de linha e coluna, respectivamente.

2.3.1.2 Rotações Discretas

As rotações discretas em múltiplos de 90 graus são transformações que preservam a estrutura de pixels da imagem. Uma rotação de 90 graus no sentido anti-horário de um *patch* I para gerar $\hat{I}$ é definida por:

$$\hat{I}(i, j) = I(j, W - 1 - i) \quad (24)$$

2.3.2 Augmentações Físicas

Enquanto as augmentações perceptuais lidam com a diversidade geométrica, as augmentações físicas visam simular as variações físicas e radiométricas que ocorrem naturalmente no processo de aquisição de dados SAR. Esta abordagem, fundamentada nos princípios explorados por Li e Peng (2023) para a augmentação de imagens SAR, expõe o modelo a um conjunto de dados sintéticos que se aproxima mais da variabilidade encontrada em dados do mundo real, resultando em um modelo mais robusto e generalizável. Diferentemente das augmentações perceptuais, estas transformações são aplicadas exclusivamente à imagem de entrada, pois representam perturbações no processo de sensoriamento que não alteram a imagem alvo subjacente.

2.3.2.1 Adição de Speckle Sintético

A augmentação de speckle sintético adiciona uma camada de ruído speckle multiplicativo, simulando uma imagem adquirida sob condições de menor processamento

multilook ou maior ruído. O ruído sintético n_s é amostrado de uma distribuição Gama, que modela com precisão a estatística do speckle em intensidade. A imagem aumentada $\hat{I}$ é obtida por:

$$\hat{I} = I_{ruidosa} \cdot n_s, \text{ onde } n_s \sim \Gamma\left(L, \frac{1}{L}\right) \quad (25)$$

Isso garante que a média do ruído seja $E[n_s] = 1$ e sua variância $Var(n_s) = 1/L$. O parâmetro L (Número Equivalente de Visadas) controla a severidade do ruído, permitindo que o modelo se torne robusto a diferentes níveis de speckle.

2.3.2.2 Simulação de Deslocamento Orbital

Pequenas imprecisões nos dados de órbita do satélite podem resultar em um deslocamento translacional de alguns pixels na imagem final. A augmentação de deslocamento orbital simula este efeito aplicando uma translação rígida $(\Delta x, \Delta y)$ à imagem:

$$\hat{I}(i, j) = I_{ruidosa}(i - \Delta x, j - \Delta y) \quad (26)$$

Onde Δx e Δy são inteiros amostrados aleatoriamente de um intervalo pré-definido. Esta técnica ensina ao modelo uma robustez local a pequenos desalinhamentos geométricos.

2.3.2.3 Adição de Ruído Térmico

A augmentação de ruído térmico simula o ruído eletrônico aditivo gerado pelos componentes do receptor do satélite. Este ruído é bem modelado por uma distribuição Gaussiana de média zero, $\mathcal{N}(0, \sigma^2)$, e sua magnitude não depende do sinal de retroespalhamento. A imagem é perturbada aditivamente:

$$\hat{I} = I_{ruidosa} + \mathcal{N}(0, \sigma^2) \quad (27)$$

Expor o modelo a este tipo de ruído o torna mais robusto à degradação da relação sinal-ruído (SNR), especialmente em áreas de baixo retroespalhamento.

2.3.2.4 Erro de Calibração Radiométrica

Imprecisões podem ocorrer na cadeia de processamento que converte os dados brutos para σ^0 , resultando em um viés radiométrico global. A augmentação de erro de calibração radiométrica que simula essa perturbação é modelada como um ganho multiplicativo aleatório g , aplicado uniformemente a toda a imagem:

$$\hat{I} = I_{ruidosa} \cdot g \quad (28)$$

Onde g é amostrado de uma distribuição uniforme (*e.g.*, entre 0.9 e 1.1). Esta técnica força o modelo a aprender características relativas e texturais em vez de depender de valores de brilho absolutos.

2.3.2.5 Simulação de Atenuação Atmosférica

A augmentação de atenuação atmosférica modela o efeito da atenuação do sinal de radar ao atravessar a atmosfera, especialmente na presença de umidade (*e.g.*, vapor d'água, chuva). Este efeito tende a ser espacialmente suave e é modelado através da aplicação de uma máscara multiplicativa de baixa frequência, como uma Gaussiana 2D, $M_G(i, j)$:

$$\hat{I}(i, j) = I_{ruidosa}(i, j) \cdot M_G(i, j) \quad (29)$$

Isso ensina o modelo a lidar com variações radiométricas de larga escala que não são devidas a mudanças no terreno.

2.3.2.6 Simulação de Degradação Espacial

A augmentação de degradação espacial simula a aquisição de uma imagem com uma resolução espacial inferior. O processo é uma composição de três operações: uma convolução com um kernel gaussiano G_σ para suavização, seguida por um subamostragem (*downsampling*) e, finalmente, uma superamostragem (*upsampling*) de volta à resolução original:

$$\hat{I} = \text{Upsampling}(\text{Downsampling}(I_{ruidosa} * G_\sigma)) \quad (30)$$

Onde $*$ denota a operação de convolução. Isso força o modelo a aprender a restaurar detalhes finos a partir de entradas com bordas e texturas borradas.

2.3.2.7 Variação do Ângulo de Incidência

Pequenas variações na órbita do satélite podem levar a ligeiras mudanças no ângulo de incidência local (θ), o que impacta diretamente a intensidade do retroespalhamento. A augmentação de variação do ângulo de incidência simula este efeito, ajustando o brilho da imagem com base em um fator de correção do cosseno da variação do ângulo, modelando a mudança na área de espalhamento projetada:

$$\hat{I} = I_{ruidosa} \cdot \cos(\Delta\theta) \quad (31)$$

Onde $\Delta\theta$ é uma pequena perturbação angular amostrada aleatoriamente.

2.3.2.8 Perturbação da Resposta de Polarização

A augmentação de perturbação de resposta de polarização simula a diafonia (*crosstalk*) entre os canais de polarização ou erros no sistema de recepção, que podem causar um desbalanceamento entre as intensidades dos canais VV e VH. Esta perturbação é modelada aplicando-se fatores multiplicativos independentes, g_{vv} e g_{vh} , a cada canal de polarização:

$$\hat{I}_{VV} = g_{vv} \cdot I_{VV} \quad (32)$$

$$\hat{I}_{VH} = g_{vh} \cdot I_{VH} \quad (33)$$

Isso ensina o modelo a não depender de uma relação fixa entre as intensidades dos canais VV e VH.

2.3.2.9 Distorção Geométrica Não-Linear

A augmentação de distorção geométrica não-linear simula deformações espaciais não-lineares que podem ser causadas por variações na propagação atmosférica ou por correções topográficas imperfeitas. Diferente da augmentação de deslocamento orbital, esta augmentação aplica um campo de deslocamento vetorial suave, $\vec{v}(i, j)$, a cada pixel:

$$\hat{I}(i, j) = I_{ruidosa}(i + v_y(i, j), j + v_x(i, j)) \quad (34)$$

Onde o campo $\vec{v}$ é gerado por amostragem de um campo de fluxo de baixa resolução e sua posterior interpolação. Isso cria distorções espacialmente correlacionadas e realistas, forçando o modelo a aprender representações de características que são robustas a pequenas deformações locais.

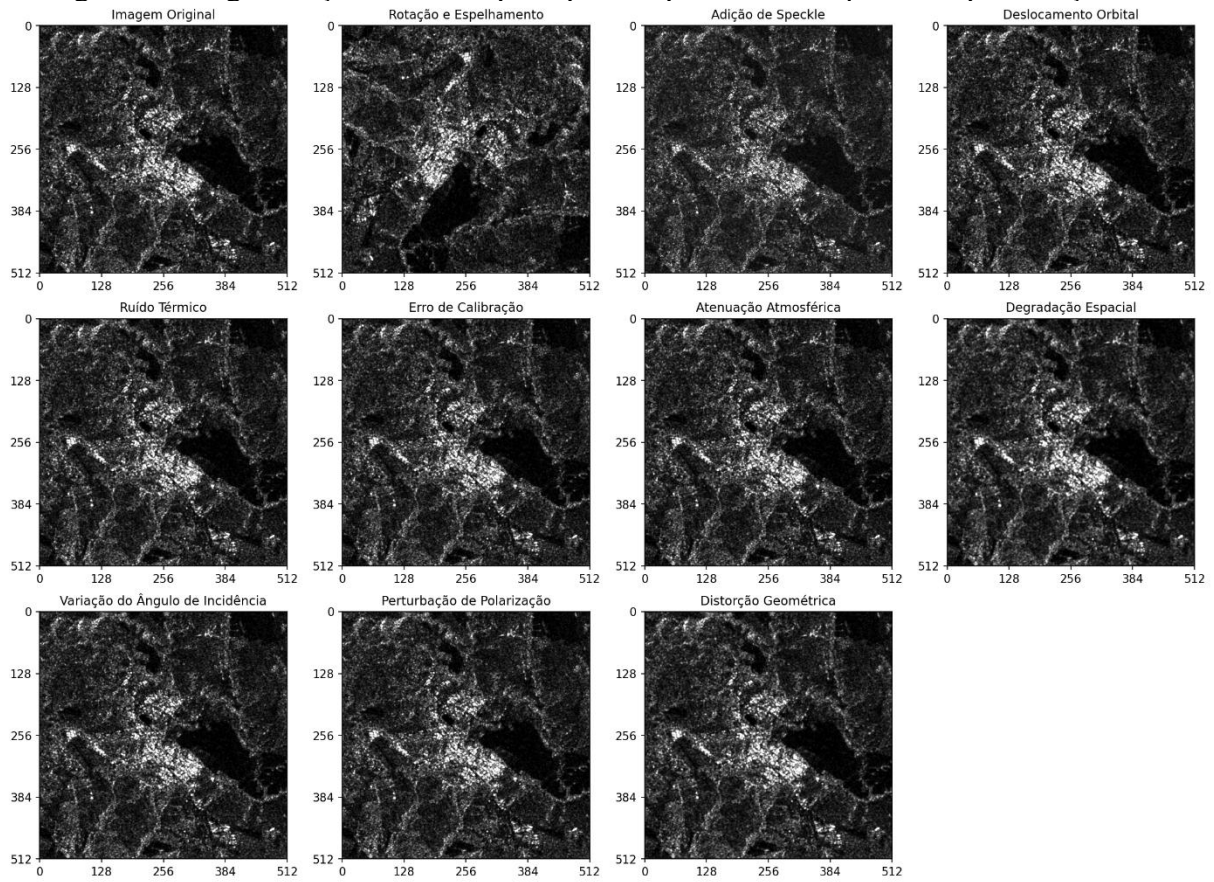
Embora as transformações individuais descritas sejam implementações de princípios físicos e de processamento de imagem conhecidos, a contribuição metodológica deste trabalho reside em dois aspectos. Primeiramente, na criação de um *framework* abrangente de nove perturbações físicas especificamente selecionadas e adaptadas para a tarefa de supressão de speckle em dados SAR. Em segundo lugar, e de forma mais significativa, no desenvolvimento de um mecanismo de aplicação probabilístico, detalhado a seguir.

2.3.2.10 Mecanismo de Aplicação Probabilístico

A aplicação das augmentações físicas não é determinística, mas sim determinada por um processo probabilístico de dois níveis. Primeiramente, para cada amostra de treinamento, uma decisão probabilística é tomada: com uma probabilidade pré-definida (**augment_probability**), o *patch* será submetido a augmentações. Esta abordagem garante que o modelo seja exposto a uma diversidade de perturbações, mas que também continue a treinar com uma maioria de amostras não alteradas, evitando um possível viés para os artefatos sintéticos.

Caso uma amostra seja selecionada para augmentação, um número aleatório de transformações, entre uma e um valor máximo definido (**max_to_apply**), é selecionado da lista de augmentações disponíveis. Essa composição aleatória de perturbações é uma contribuição central deste *framework*, pois simula de forma mais realista as condições do mundo real, onde múltiplos efeitos físicos podem ocorrer simultaneamente. Este mecanismo força o modelo a aprender a dissociar o sinal do ruído sob uma variedade muito maior de condições, promovendo uma regularização mais forte e uma maior capacidade de generalização.

Figura 7 – Augmentações físicas e perceptuais aplicadas à um *patch* na polarização VH



Fonte: Autoria Própria.

Ao ser treinado com essa diversidade de perturbações realistas, o modelo aprende a dissociar o sinal SAR subjacente de um amplo espectro de artefatos, melhorando significativamente sua robustez e capacidade de generalização para dados do mundo real.

2.4 MÉTRICAS PARA AVALIAÇÃO DE QUALIDADE DE IMAGEM

A avaliação de um algoritmo de restauração de imagens, como um filtro de supressão de speckle, não pode depender unicamente da subjetividade da inspeção visual. Para realizar uma análise objetiva, rigorosa e comparável, é indispensável o uso de métricas de avaliação quantitativas. Essas métricas traduzem a qualidade de uma imagem em um valor numérico, permitindo a comparação direta do desempenho entre diferentes métodos e configurações. As métricas de qualidade de imagem são geralmente divididas em duas categorias principais, ambas exploradas neste trabalho: as métricas referenciais, que comparam a imagem restaurada com uma *ground truth* ideal, e as métricas não-referenciais, que avaliam a qualidade da imagem

com base em suas próprias características estatísticas, sem a necessidade de uma imagem de referência.

2.4.1 Métricas Referenciais

As métricas referenciais (*full-reference metrics*) são utilizadas quando uma imagem "limpa" ou "ideal" está disponível para servir como base de comparação com a imagem restaurada. Embora no mundo real uma imagem SAR perfeitamente livre de speckle não exista, a imagem-alvo gerada por filtragem multitemporal serve como uma excelente aproximação para esta finalidade.

2.4.1.1 Peak Signal-to-Noise Ratio (PSNR)

Uma das métricas mais tradicionais, a Razão Sinal-Ruído de Pico (*Peak Signal-to-Noise Ratio* - PSNR) mede a razão entre a máxima potência possível de um sinal e a potência do erro que o corrompe. É inversamente proporcional ao Erro Quadrático Médio (MSE), definido como:

$$MSE = \frac{1}{H \times W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} [I_{alvo}(i,j) - \hat{I}(i,j)]^2 \quad (35)$$

O PSNR é então calculado em uma escala logarítmica (decibéis - dB):

$$PSNR = 10 \cdot \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \quad (36)$$

Onde MAX_I é o valor máximo possível para um pixel na imagem. Um valor de PSNR mais alto indica uma menor distorção e uma maior fidelidade dos pixels da imagem restaurada em relação à original.

2.4.1.2 Structural Similarity Index (SSIM)

O Índice de Similaridade Estrutural (*Structural Similarity Index* – SSIM) (WANG *et al.*, 2004) foi projetado para superar a principal limitação do PSNR, que é sua baixa correlação com a percepção visual humana. Em vez de avaliar o erro pixel a pixel, o SSIM compara janelas locais de duas imagens com base em três componentes: luminância (l), contraste (c) e estrutura (s). O SSIM entre duas janelas x e y é definido como:

$$SSIM(x, y) = [l(x, y)]^\alpha \cdot [c(x, y)]^\beta \cdot [s(x, y)]^\gamma \quad (37)$$

onde α, β, γ são pesos (geralmente definidos como 1). O valor final da métrica é a média dos valores de SSIM de todas as janelas. O SSIM varia de -1 a 1, onde 1 indica identidade perfeita. Um valor mais próximo de 1 significa que a imagem restaurada preservou melhor a estrutura perceptual do alvo.

2.4.1.3 Edge Preservation Degree (EPD)

O Grau de Preservação de Bordas (*Edge Preservation Degree* – EPD) é uma métrica que avalia a capacidade de um filtro de preservar as bordas e contornos presentes na imagem original. Ela é calculada como o coeficiente de correlação entre os mapas de gradiente (obtidos com um operador de Sobel) da imagem alvo (G_{alvo}) e da imagem restaurada ($\hat{G}$). Um valor de EPD mais próximo de 1 indica uma excelente preservação das bordas.

2.4.1.4 Pratt's Figure of Merit (FOM)

O FOM (PRATT, 2007) é uma métrica mais rigorosa para avaliar a qualidade da detecção de bordas. Ela compara os pixels de borda detectados na imagem restaurada com os da imagem alvo, penalizando tanto a ausência de bordas verdadeiras quanto a presença de bordas falsas, além de penalizar deslocamentos das bordas corretas. Um valor de FOM mais próximo de 1 indica uma detecção de bordas mais precisa e fiel.

2.4.1.5 Distância de Kolmogorov-Smirnov (KS)

A Distância de Kolmogorov-Smirnov (KS) é uma métrica não paramétrica que quantifica a distância entre as funções de distribuição acumulada (FDA) empíricas de duas amostras. No contexto de imagens, ela compara a distribuição global de intensidades de pixels da imagem restaurada com a da imagem alvo. A estatística KS, D , é definida como a distância vertical máxima entre as duas FDAs:

$$D_{KS}(F_{alvo}, F_I) = \sup_x |F_{alvo}(x) - F_I(x)| \quad (38)$$

Onde $F(x)$ é a FDA e $\sup_x$ é o supremo do conjunto de distâncias. Um valor de D mais baixo (próximo de zero) indica que as duas imagens possuem distribuições de pixels estatisticamente mais similares.

2.4.1.6 Divergência de Kullback-Leibler (KL)

A Divergência de Kullback-Leibler (KL), ou entropia relativa (KULLBACK; LEIBLER, 1951), mede como uma distribuição de probabilidade se afasta de uma segunda distribuição de referência. Para imagens, os histogramas normalizados são tratados como distribuições de probabilidade discretas (P para o alvo e Q para a imagem restaurada). A divergência de Q a partir de P é:

$$D_{KL}(P||Q) = \sum_i P(i) \cdot \log\left(\frac{P(i)}{Q(i)}\right) \quad (39)$$

Um valor de DKL mais baixo (próximo de zero) indica que a radiometria geral da imagem restaurada foi bem preservada em relação ao alvo.

2.4.1.7 Entropia de Shannon

A Entropia de Shannon (SHANNON, 1948) quantifica a incerteza ou a "quantidade de informação" em uma fonte de dados. Para uma imagem, ela é calculada a partir do histograma de seus níveis de cinza, medindo a complexidade global da imagem:

$$H = - \sum_i p_i \cdot \log_2(p_i) \quad (40)$$

Onde p_i é a probabilidade do i -ésimo nível de cinza. Embora não seja uma métrica de comparação direta, a comparação da entropia da imagem restaurada com a do alvo pode indicar se o filtro está removendo informação (texturas), resultando em uma entropia mais baixa, ou introduzindo artefatos, resultando em uma entropia mais alta.

2.4.2 Métricas Não-Referenciais

As métricas não-referenciais (*no-reference metrics*), também conhecidas como "métricas cegas" (*blind metrics*), são essenciais para a avaliação de filtros de speckle, pois permitem quantificar a qualidade de uma imagem sem a necessidade de uma *ground truth*. Elas são particularmente úteis para comparar a eficácia dos filtros multitemporais, onde a qualidade é medida pela conformidade da imagem resultante com as propriedades estatísticas esperadas de uma imagem SAR "limpa".

2.4.2.1 Equivalent Number of Looks (ENL)

O Número Equivalente de Visadas (*Equivalent Number of Looks* - ENL) é a métrica mais importante para quantificar a eficácia de um filtro na supressão do ruído speckle. Ele mede a suavidade de uma região que deveria ser homogênea. Para uma área com média μ e desvio padrão σ , o ENL é definido como:

$$ENL = \left(\frac{\mu}{\sigma} \right)^2 \quad (41)$$

Em uma imagem SAR, uma região homogênea (*e.g.*, um campo agrícola) idealmente teria um retroespalhamento constante, mas o speckle introduz uma variação estatística. Um filtro eficaz aumenta a relação sinal-ruído, resultando em um desvio padrão menor para a mesma média. Portanto, um valor de ENL mais alto indica uma supressão de speckle mais eficaz.

2.4.2.2 Edge Preservation Index (EPI)

O Índice de Preservação de Bordas (*Edge Preservation Index* - EPI) é semelhante ao EPD, mas não-referencial, o EPI mede a preservação das bordas comparando a imagem filtrada ($\hat{I}$) com a imagem ruidosa original ($I_{ruidosa}$). É calculado como o coeficiente de correlação entre os mapas de gradiente de alta frequência das duas imagens. Um valor de EPI mais próximo de 1 indica que o filtro removeu o ruído sem borrar ou destruir as bordas e estruturas presentes na imagem original.

2.4.2.3 Razão das Médias (Ratio of Means)

A Razão das Médias (*Ratio of Means*) é uma métrica que avalia a preservação radiométrica do filtro. É calculada como a razão entre a média da imagem filtrada e a média da imagem original:

$$Ratio = \frac{\mu_{\hat{I}}}{\mu_{I_{ruidosa}}} \quad (42)$$

Um filtro ideal deve apenas suavizar o ruído sem alterar o brilho médio da cena. Portanto, um valor de *Ratio of Means* mais próximo de 1.0 é desejável, indicando que a radiometria geral foi preservada.

2.4.2.4 Speckle Index

O índice de Speckle (*Speckle Index*) é o coeficiente de variação calculado em janelas locais sobre a imagem:

$$Speckle Index = \frac{\sigma}{\mu} \quad (43)$$

Ele mede o ruído residual na imagem filtrada. Um valor de *Speckle Index* mais baixo indica uma supressão de ruído mais eficaz.

2.4.2.5 Matriz de Coocorrência de Níveis de Cinza (GLCM)

A Matriz de Coocorrência de Níveis de Cinza (*Gray-Level Co-occurrence Matrix* - GLCM) é uma matriz que descreve a frequência com que diferentes combinações de níveis de cinza ocorrem em pares de pixels a uma determinada distância e orientação. A partir dela, métricas de textura de segunda ordem podem ser extraídas (HARALICK; SHANMUGAM; DINSTEN, 1973). As duas principais utilizadas neste trabalho são a Homogeneidade, que mede a suavidade da textura local (um valor mais próximo do valor original da imagem de referência é melhor), e o Contraste, que mede a variação local (um valor mais baixo é melhor, quando acompanhado da melhora em outras métricas, como o ENL, mas valores excessivamente baixos podem indicar o borramento de texturas reais).

2.4.2.6 Entropia Local (Local Entropy)

Similar à Entropia de Shannon, a Entropia Local mede a complexidade da textura, mas o faz em janelas locais sobre a imagem. O valor final é a média das entropias de todas as janelas. Um filtro eficaz deve reduzir a entropia local ao suavizar o ruído em áreas homogêneas, mas sem reduzi-la excessivamente em áreas texturizadas. Um valor de entropia local mais baixo geralmente indica uma suavização mais eficaz.

3 MATERIAIS E MÉTODOS

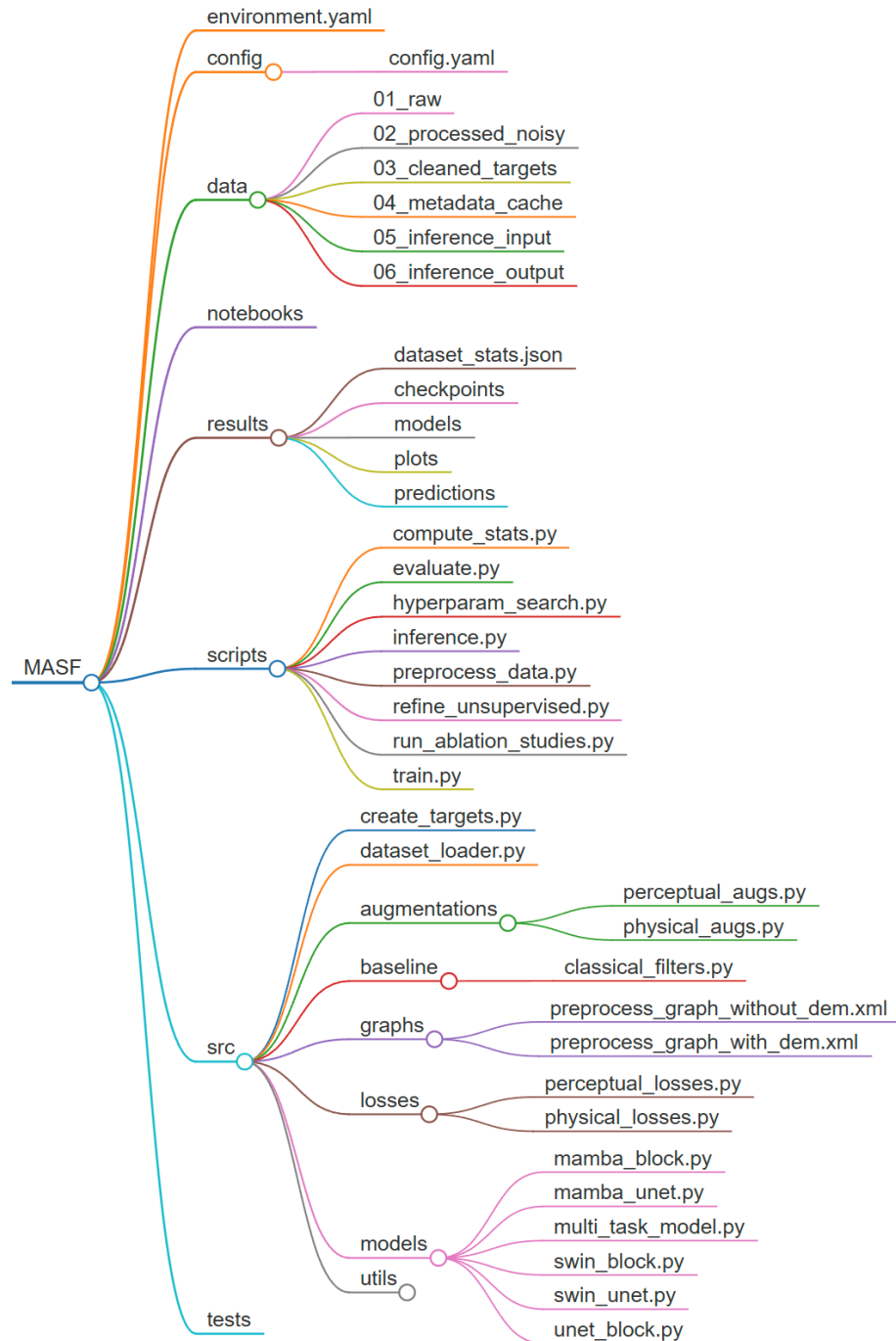
Este capítulo detalha a implementação do MASF (APÊNDICE C – REPOSITÓRIO DO CÓDIGO-FONTE DO MASF), a metodologia desenvolvida e empregada neste trabalho para a supressão de ruído speckle em imagens SAR. Enquanto o capítulo anterior estabeleceu a fundamentação teórica dos conceitos de sensoriamento remoto, *Deep Learning* e avaliação de imagem, este capítulo se aprofunda na implementação da arquitetura do software, do *pipeline* de processamento de dados, do design do modelo e do método experimental utilizado para validar a proposta. Cada seção aborda um componente lógico do *framework*, fundamentando sua implementação no código-fonte do projeto à teoria apresentada no capítulo anterior e justificando as escolhas de engenharia e design que garantem a robustez e reprodutibilidade dos resultados.

3.1 ESTRUTURA DO FRAMEWORK

A reprodutibilidade é fundamental na pesquisa científica. No contexto da ciência da computação e do *Deep Learning*, garantir que um experimento possa ser replicado de forma idêntica por outros pesquisadores é essencial para a sua validação. Com este princípio em mente, o MASF foi projetado não apenas como um modelo, mas como um *software* modular e configurável, cuja estrutura visa maximizar a clareza, a edição, a manutenção e, principalmente, a reprodutibilidade.

A organização do projeto segue uma estrutura de diretórios lógica e padronizada, onde cada pasta possui uma responsabilidade bem definida, separando dados, código-fonte, configurações e resultados. Esta separação é necessária para a gestão de um projeto de pesquisa complexo e multifacetado.

Figura 8 – Diagrama em árvore da estrutura de diretórios e arquivos principais do MASF



Fonte: Autoria Própria.

A seguir, uma descrição da função de cada diretório principal:

a) **/config**: Este diretório centraliza todos os parâmetros e hiperparâmetros do projeto em um único arquivo, **config.yaml**, bem como em um arquivo de testes **test_config.yaml**. A utilização de um arquivo de configuração externo desacopla a lógica do código dos parâmetros experimentais. Isso permite que qualquer aspecto do *pipeline*, desde os caminhos dos arquivos e o número de épocas de treinamento até os pesos das funções de perda e os parâmetros das

augmentações, seja modificado sem a necessidade de alterar o código-fonte. Esta abordagem é crucial para a reprodutibilidade, pois um único arquivo de configuração pode ser salvo e compartilhado para replicar um experimento em sua totalidade;

b) **/data**: Contém todos os dados utilizados e gerados pelo *pipeline*, organizados em subdiretórios que representam cada estágio do processamento, desde os dados brutos na pasta **/01_raw** até os resultados da inferência na pasta **/06_inference_output**. Esta estrutura padronizada garante a organização e o rastreamento dos dados ao longo de todo o fluxo de processamento;

c) **/src**: Contém toda a lógica de programação em *Python*. Este diretório é subdividido em módulos que encapsulam funcionalidades específicas: **/models** (definições das arquiteturas de rede), **/losses** (implementação das funções de perda), **/utils** (funções auxiliares), entre outros. Essa modularidade promove o reuso de código e facilita a manutenção e implementação de novos recursos no *framework*;

d) **/scripts**: Contém os *scripts* executáveis de alto nível que coordenam o *pipeline*. Cada *script* é responsável por uma macroetapa do processo, lendo a configuração do arquivo **config.yaml** e chamando as funções necessárias do diretório **/src**. Essa separação entre os scripts de coordenação e a lógica de baixo nível simplifica o fluxo de trabalho e o torna fácil de entender;

e) **/results**: Um diretório dedicado ao armazenamento de todos os arquivos gerados durante os experimentos, como os modelos treinados em **/models**, os gráficos de análise em **/plots**, os *checkpoints* do treinamento em **/checkpoints** e as tabelas de métricas. Manter os resultados separados dos dados e do código evita a desorganização e facilita a análise e a comparação entre diferentes execuções;

f) **/tests**: Contém uma suíte de testes de unidade e de integração, utilizando o *framework* *pytest*. A presença de testes automatizados garante a confiabilidade e a funcionalidade de cada componente do sistema, desde as funções de perda até os *pipelines* de treinamento completos.

Portanto, a estrutura do MASF adota os princípios da engenharia de software para construir um ambiente de pesquisa onde a clareza, a modularidade e a automação são empregadas para garantir que os resultados obtidos não sejam apenas de alta qualidade, mas também cientificamente válidos e reprodutíveis.

3.2 PIPELINE DE PRÉ-PROCESSAMENTO DE DADOS

A transformação de dados brutos de satélite em um formato adequado para o treinamento de modelos de *Deep Learning* é uma etapa fundamental que impacta diretamente a qualidade e a confiabilidade dos resultados. O MASF emprega um pipeline de pré-processamento semiautomatizado, dividido em três estágios principais: a correção radiométrica e geométrica utilizando o software ESA SNAP, o correto registro da série temporal através de um algoritmo customizado, e a geração de imagens-alvo "limpas" por meio de filtragem multitemporal. Esta seção detalha a metodologia e a justificativa técnica por trás de cada uma dessas etapas.

3.2.1 Área de Estudo e Conjunto de Dados Sentinel-1

A capacidade de um modelo de *Deep Learning* de generalizar para dados não vistos depende da diversidade e da representatividade do seu conjunto de treinamento. Conforme argumentado por Bauer-Marschallinger *et al.* (2019), modelos treinados em uma gama restrita de condições de terreno podem falhar ao serem aplicados em novos ambientes, pois aprendem a otimizar para as características específicas do conjunto de treino. Para mitigar esse risco e desenvolver um modelo de supressão de speckle que seja robusto e aplicável globalmente, este trabalho utiliza um conjunto de dados diversificado, composto por imagens Sentinel-1 GRD coletadas sobre dez localidades distintas, abrangendo uma vasta gama de biomas, relevos e condições de cobertura do solo.

Um critério fundamental para a seleção dessas áreas foi a estabilidade topográfica temporal. A geração de um alvo multitemporal "limpo" pressupõe que as variações no retroespalhamento ao longo da série temporal sejam devidas principalmente ao speckle e a mudanças sazonais diminutas (*e.g.*, umidade, crescimento de vegetação), e não a alterações estruturais drásticas no terreno, como desmatamento ou expansão urbana. A literatura demonstra que ambientes com baixa intervenção antrópica, como desertos hiperáridos, apresentam alta estabilidade temporal no sinal de retroespalhamento SAR, tornando-os alvos ideais para a calibração e validação de modelos (ULLMANN *et al.*, 2019). O Quadro 1 detalha as localizações escolhidas, que foram agrupadas por tipo de bioma e relevo para garantir uma amostragem representativa de alvos estáveis.

Quadro 1 – Descrição das áreas de estudo

Localização (País)	Bioma Predominante	Características Relevantes
Deserto de Atacama (Chile)	Deserto Hiperárido	Relevo montanhoso, ausência de vegetação, alvos geológicos estáveis.
Deserto de Negev (Israel)	Deserto Semiárido	Relevo variado com planaltos e uádís, vegetação esparsa.
Deserto de Sonora (EUA)	Deserto Subtropical	Cactos e vegetação adaptada à aridez, relevo de bacias e montanhas.
Outback Australiano (Austrália)	Deserto e Matagal Xérico	Extensas planícies, baixa densidade de vegetação, solos expostos.
Tóquio (Japão)	Urbano Denso	Alta concentração de espalhadores persistentes (edifícios, infraestrutura).
Mar de Aral (Uzbequistão)	Deserto e Estepe	Leito de lago seco com alta salinidade, áreas de estepe.
Patagônia, Chubut (Argentina)	Estepe Fria e Deserto	Planaltos, vegetação de estepe, ventos fortes, baixa umidade.
Koue Bokkeveld (África do Sul)	Fynbos e Agricultura	Mosaico de vegetação nativa (Fynbos) e pomares, relevo montanhoso.
Rocky Mountain, Colorado (EUA)	Floresta de Coníferas e Tundra Alpina	Relevo montanhoso, florestas perenes, áreas de tundra.
Estreito de Gibraltar (Marrocos)	Vegetação Mediterrânea	Relevo variado, mosaico de florestas, matagais, áreas agrícolas e urbanas.

Fonte: Autoria Própria.

A seleção de desertos (Atacama, Negev, Outback, Sonora) e estepes áridas (Sahel) é fundamentada em estudos como o de Ullmann *et al.* (2019), que demonstram a excepcional estabilidade radiométrica de longo prazo de regiões áridas em dados de Banda-C, atribuída à baixa umidade do solo e à ausência de vegetação. Para o Deserto de Sonora, especificamente,

sua estabilidade superficial é corroborada por seu uso como um sítio de calibração pseudo-invariante para múltiplos sensores (ANGAL, A. *et al.*, 2010). Para garantir a diversidade e a generalização do modelo, as demais localidades foram selecionadas com base no mapa global de coerência interferométrica de Kelldorfer *et al.* (2022). Este estudo justifica a inclusão de áreas que representam um espectro variado de estabilidade temporal, como a região metropolitana de Tóquio, onde a alta coerência é mantida por estruturas urbanas estáveis, e outras paisagens com dinâmicas sazonais. A inclusão de uma região de cobertura mista em Marrocos é validada por El Mortaji, N. *et al.* (2022), que identificaram alta estabilidade em classes de uso do solo como florestas, áreas construídas e solo nu, complementando a variedade do conjunto de dados.

Esta seleção heterogênea garante que o modelo seja treinado com uma ampla variedade de assinaturas de retroespalhamento, forçando-o a aprender a verdadeira estrutura estatística do ruído speckle em vez de memorizar as texturas de um tipo específico de paisagem, aumentando sua capacidade de generalização. Os produtos associados às localizações escolhidas se encontram no APÊNDICE A – Produtos do Sentinel-1 Utilizados.

3.2.2 Correção Radiométrica e Geométrica com ESA SNAP

O ponto de partida do *pipeline* são os produtos Sentinel-1 Nível-1 GRD, que, embora já projetados em coordenadas terrestres, ainda contêm distorções geométricas e radiométricas que precisam ser corrigidas para garantir a comparabilidade entre imagens e a física dos dados. Para isso, o *framework* utiliza a Ferramenta de Processamento de Grafos (GPT) do Sentinel Application Platform (SNAP), fornecido pela ESA.

O processo é coordenado pelo script **scripts/preprocess_data.py**, que executa de forma paralela os grafos XML definidos em **src/graphs/**. A sequência de operações ideal, como sugerida por Filipponi (2019), é a seguinte:

a) Aplicação do Arquivo de Órbita (*Apply-Orbit-File*): Os metadados de órbita que acompanham os produtos GRD não possuem a precisão necessária para um georreferenciamento preciso. Esta etapa substitui os metadados pelos arquivos de órbita precisa (*Precise Orbit Ephemerides* - POE), que são disponibilizados pela ESA dias após a aquisição e contêm informações de posicionamento do satélite com precisão, sendo um pré-requisito para o correto registro de alta qualidade;

b) Remoção de Ruído Térmico (*ThermalNoiseRemoval*): O ruído térmico, gerado pelos componentes eletrônicos do receptor do satélite, manifesta-se como um ruído aditivo que pode

degradar a qualidade da imagem, principalmente em áreas de baixo retroespalhamento, como corpos d'água. Esta etapa subtrai o perfil de ruído, que é modelado e fornecido nos metadados do produto;

c) Remoção de Ruído de Borda do GRD (*Remove-GRD-Border-Noise*): As bordas das faixas de aquisição do Sentinel-1 frequentemente contêm pixels com valores inválidos ou de baixa qualidade devido à queda de ganho da antena. Este operador identifica e cria uma máscara para esses pixels, evitando que afetem as etapas posteriores;

d) Calibração Radiométrica: Este é o passo mais crítico para a análise quantitativa. Ele converte os valores de número digital (DN) brutos, que não possuem significado físico, para uma medida fisicamente interpretável do retroespalhamento do radar, o Sigma Naught (σ^0). O σ^0 representa a refletividade por unidade de área no plano do solo, normalizada pelo ângulo de incidência local, o que o torna a medida ideal para a comparação de alvos em diferentes imagens e condições de aquisição;

e) Correção do Elipsoide e Adição de Dados Auxiliares: Utilizando um Modelo Digital de Elevação (MDE), como o SRTM de 1 segundo de arco, o grafo realiza a correção geométrica final (ortorretificação), removendo distorções causadas pelo relevo, como *foreshortening* e *layover*. Durante este processo, o grafo **preprocess_graph_with_dem.xml** extrai e salva como bandas adicionais o ângulo de incidência e a elevação, da qual se obtém a inclinação do terreno (*slope*), que servirão como canais de entrada auxiliares para o modelo de *Deep Learning*.

Ao final desta etapa, os arquivos (.zip) brutos são convertidos em produtos no formato BEAM-DIMAP, prontos para o corregristo.

3.2.3 Corregristo da Série Temporal

O corregristo é o processo de alinhar geometricamente uma série de imagens para que cada pixel (i, j) corresponda à mesma localização geográfica em todas as imagens da pilha. Um corregristo de alta precisão é necessário para a filtragem multitemporal, pois erros de alinhamento de apenas um pixel podem ser interpretados erroneamente como ruído ou como mudanças temporais, degradando a qualidade da imagem-alvo.

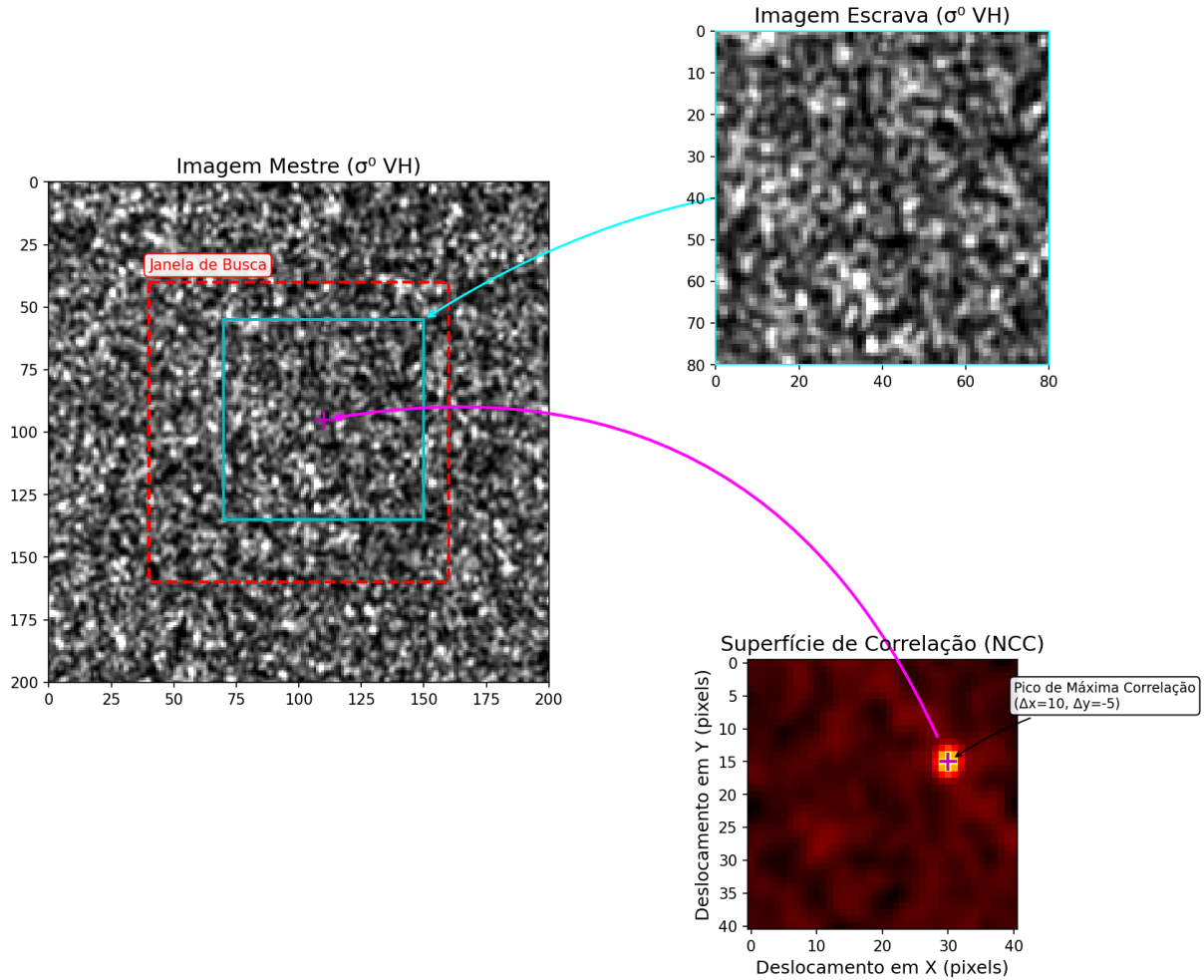
Embora o SNAP possua ferramentas para esta finalidade, testes preliminares realizados durante o desenvolvimento deste trabalho revelaram que seu desempenho para grandes volumes de dados é computacionalmente proibitivo, com o tempo de processamento escalando de forma desfavorável com o número de imagens. Para contornar essa limitação mantendo controle total sobre o processo, foi desenvolvido um algoritmo de corregristo customizado, implementado no

script `src/create_targets.py`, que emprega ferramentas geoespaciais consolidadas e técnicas de processamento de imagem. O processo é dividido em duas fases:

a) Seleção da Imagem Mestre e Corregistro Grosso: Para garantir a consistência geométrica da pilha de imagens, uma única imagem da série temporal é selecionada como a referência ("mestre"). A seleção prioriza a imagem que contém os dados auxiliares de elevação, pois ela serve como a base geométrica mais completa. Todas as outras imagens ("escravas") são então alinhadas à grade da imagem mestre através da função *Warp*, da biblioteca GDAL. Esta etapa, conhecida como corregistro grosso, projeta todas as imagens para o mesmo sistema de coordenadas, resolução e extensão, corrigindo as maiores discrepâncias de posicionamento e garantindo uma sobreposição inicial consistente;

a) Corregistro Fino por Correlação Cruzada Normalizada (NCC): Mesmo após o alinhamento grosso, desalinhamentos residuais de nível subpixel, causados por pequenas variações orbitais ou atmosféricas, podem persistir. Para corrigi-los, o algoritmo implementa um corregistro fino baseado em área, utilizando a técnica de Correlação Cruzada Normalizada (NCC). A imagem é dividida em uma grade de blocos, e para cada bloco da imagem escrava, uma janela de busca maior é definida na imagem mestre. A NCC é então calculada entre o bloco e a janela de busca, gerando um mapa de correlação cujo pico indica a posição de máxima similaridade. Este pico revela o deslocamento residual (Δx , Δy) entre as duas imagens naquela região específica. Para alcançar uma precisão de subpixel, uma interpolação parabólica é aplicada em torno do pico de correlação, refinando a estimativa do deslocamento. Este processo é repetido para toda a grade, gerando um campo de deslocamento que é subsequentemente usado para deformar as imagens escravas, alinhando-as perfeitamente à imagem mestre.

Figura 9 – Representação da janela de busca e superfície de correlação



Fonte: Autoria Própria.

Ao final deste processo, obtém-se um conjunto de imagens de série temporal perfeitamente alinhadas, salvas no formato GeoTIFF e prontas para a etapa de geração de alvos.

3.2.4 Geração de Alvos Multitemporais

A geração de uma imagem-alvo "limpa" (*ground truth*) é um dos maiores desafios no treinamento supervisionado de modelos de remoção de speckle, pois uma imagem SAR nunca está livre de ruído. A abordagem adotada neste trabalho, e que fundamenta o componente "Multitemporal" do *framework*, é a utilização da redundância de informação presente em uma série temporal corregistrada para reduzir o speckle de forma estatística.

Conforme detalhado no Capítulo 2, a média de N imagens independentes reduz o desvio padrão do ruído speckle por um fator de $1 / \sqrt{N}$. Para determinar o método de filtragem multitemporal mais eficaz para este trabalho, foi realizada uma análise comparativa entre as

três abordagens implementadas em `src/utills/temporal_filters_utils.py`: Média Simples, Quegan e Stroobants. Utilizando o script `scripts/compare_multitemporal_filters.py`, foram calculadas métricas de qualidade de imagem sem referência para a imagem original e pilhas de 3, 6 e 9 imagens, comparando-as também com filtros espaciais clássicos para as polarizações VV e VH. A Tabela 1 e a Tabela 2 apresentam um resumo dos resultados obtidos.

Tabela 1 – Análise quantitativa de filtragem na polarização VV

N	Filtro (Filter)	Ratio of Means	Speckle Index	EPI	ENL	Local Entropy	GLCM Homogeneity	GLCM Contrast
1	Original	1,000	0,341	1,000	2,621	3,013	0,420	84,247
	Lee	1,004	0,213	0,952	4,218	2,434	0,565	35,813
	Frost	1,002	0,244	0,338	4,594	2,690	0,509	31,664
	Gamma MAP	0,994	0,151	0,534	7,703	2,737	0,524	20,861
	Lee Sigma	0,943	0,164	0,603	7,500	2,895	0,491	28,605
3	Média (Mean)	0,986	0,292	0,750	4,528	2,866	0,451	57,533
	Stroobants	0,985	0,292	0,754	4,530	2,878	0,450	56,443
	Quegan	0,986	0,292	0,750	4,528	2,866	0,451	57,533
6	Média (Mean)	0,993	0,267	0,681	6,161	2,873	0,452	62,057
	Stroobants	0,992	0,267	0,673	6,190	2,873	0,452	62,757
	Quegan	0,993	0,267	0,681	6,161	2,873	0,452	62,057
9	Média (Mean)	1,003	0,248	0,644	7,089	2,866	0,456	56,961
	Stroobants	1,001	0,248	0,637	7,120	2,847	0,459	56,970
	Quegan	1,003	0,248	0,644	7,089	2,866	0,456	56,961

Fonte: Autoria Própria.

Tabela 2 – Análise quantitativa de filtragem na polarização VH

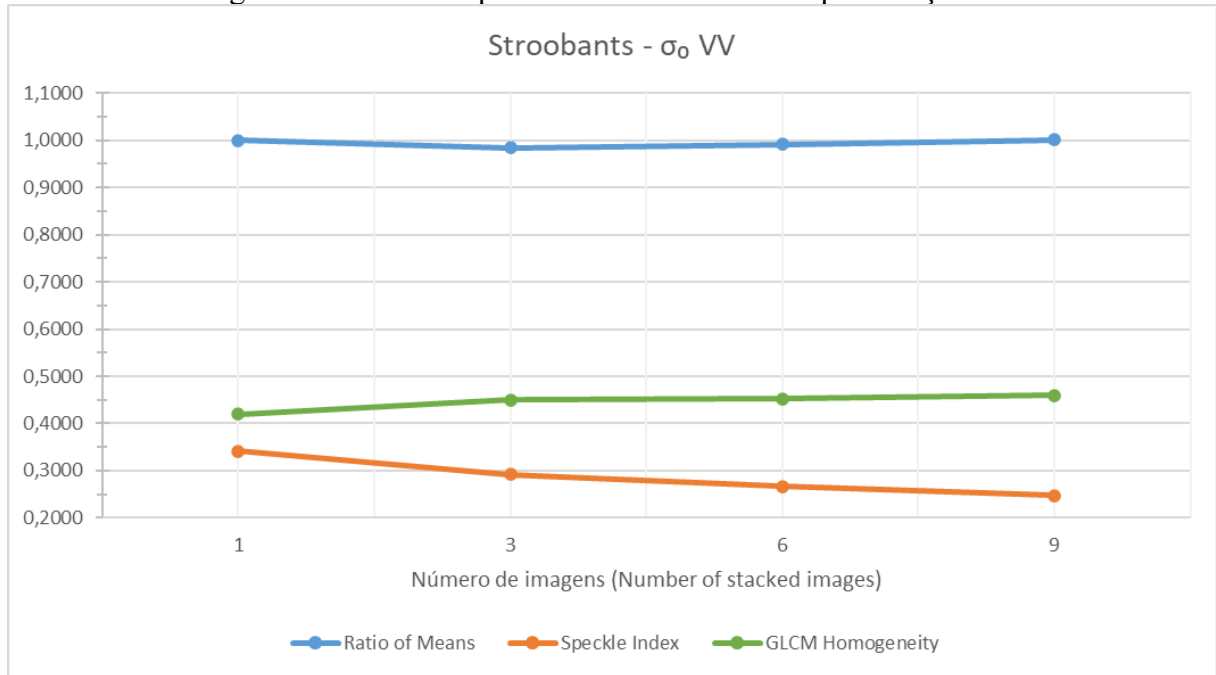
N	Filtro (Filter)	Ratio of Means	Speckle Index	EPI	ENL	Local Entropy	GLCM Homogeneity	GLCM Contrast
1	Original	1,000	0,443	1,000	2,273	3,250	0,372	101,83
	Lee	1,005	0,278	0,947	3,790	2,667	0,521	43,462
	Frost	1,006	0,324	0,329	3,980	2,983	0,448	40,741
	Gamma MAP	0,981	0,197	0,534	6,923	3,002	0,472	27,013
	Lee Sigma	0,928	0,195	0,580	6,663	3,102	0,455	28,901
3	Média (Mean)	1,002	0,326	0,692	4,341	3,164	0,399	89,886
	Stroobants	1,002	0,326	0,693	4,331	3,139	0,404	89,884
	Quegan	1,002	0,326	0,692	4,341	3,164	0,399	89,886
6	Média (Mean)	1,021	0,282	0,607	6,021	3,104	0,411	74,094
	Stroobants	1,021	0,282	0,607	6,017	3,094	0,413	74,066
	Quegan	1,021	0,282	0,607	6,021	3,104	0,411	74,094
9	Média (Mean)	1,057	0,260	0,563	6,822	3,093	0,415	66,066
	Stroobants	1,057	0,260	0,563	6,823	3,083	0,417	66,027
	Quegan	1,057	0,260	0,563	6,822	3,093	0,415	66,066

Fonte: Autoria Própria.

A análise das métricas das tabelas acima revela uma complexa relação entre os diferentes filtros. Uma avaliação superficial poderia sugerir a superioridade dos filtros Gamma MAP e Lee Sigma, que apresentam os menores valores de *Speckle Index* e os maiores de ENL, indicando uma supressão de ruído mais agressiva. Contudo, essa eficácia vem ao custo de uma degradação significativa de outras características da imagem: seus valores de *GLCM Contrast* são os mais baixos, o EPI é apenas moderado, e a Razão das Médias se desvia do valor ideal (1,0), indicando uma suavização excessiva que remove não apenas o ruído, mas também a textura e a fidelidade radiométrica da cena. Em contraste, o filtro Lee, embora menos eficaz na remoção de ruído, destaca-se por manter uma excelente preservação radiométrica e de bordas (Razão das Médias e EPI próximos de 1,0), mas falha em suavizar áreas homogêneas, resultando na maior *GLCM Homogeneity* e embora apresente o maior *GLCM Contrast*, ainda é consideravelmente inferior aos filtros multitemporais nesta métrica. O filtro Frost se apresenta como um intermediário, com uma boa preservação radiométrica, mas com perdas significativas na preservação de bordas.

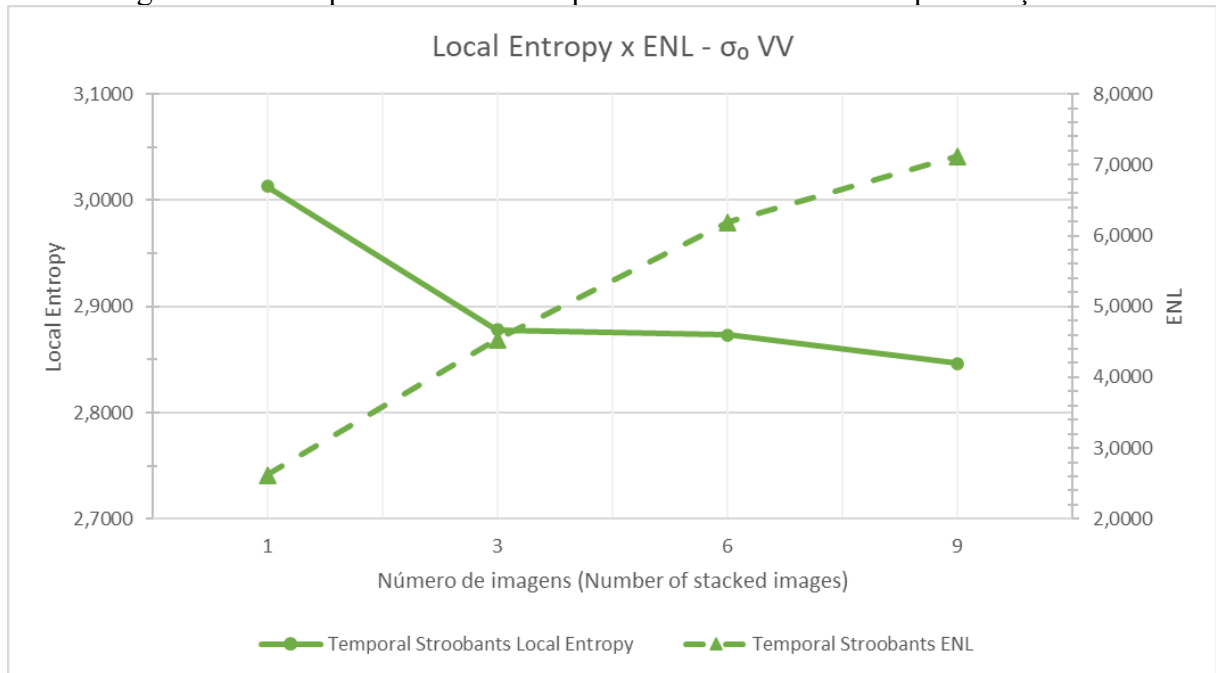
A distinção entre os métodos multitemporais, por sua vez, foi avaliada através da análise de seu comportamento em função do número de imagens na série temporal. Embora os três filtros (Média, Quegan e Stroobants) apresentem resultados muito semelhantes, Stroobants demonstrou consistentemente o balanço mais favorável. Sua abordagem de ponderar a média com base na distância à mediana temporal confere-lhe uma maior robustez a *outliers*, imagens na série que possam conter mudanças sazonais ou de umidade do solo, permitindo uma supressão de ruído eficaz sem sacrificar a integridade de bordas e texturas finas. Isso foi corroborado por valores de ENL e *GLCM Contrast* marginalmente superiores aos dos outros filtros multitemporais e, principalmente, o melhor desempenho geral na métrica de Razão das Médias para a polarização VV. É importante notar, contudo, que todos os filtros multitemporais apresentaram um pequeno desvio radiométrico na polarização VH de 5,7% acima do valor ideal de 1,0.

Figura 10 – Métricas para o filtro Stroobants na polarização VV



Fonte: Autoria Própria.

Figura 11 –Entropia Local vs. ENL para o filtro Stroobants na polarização VV

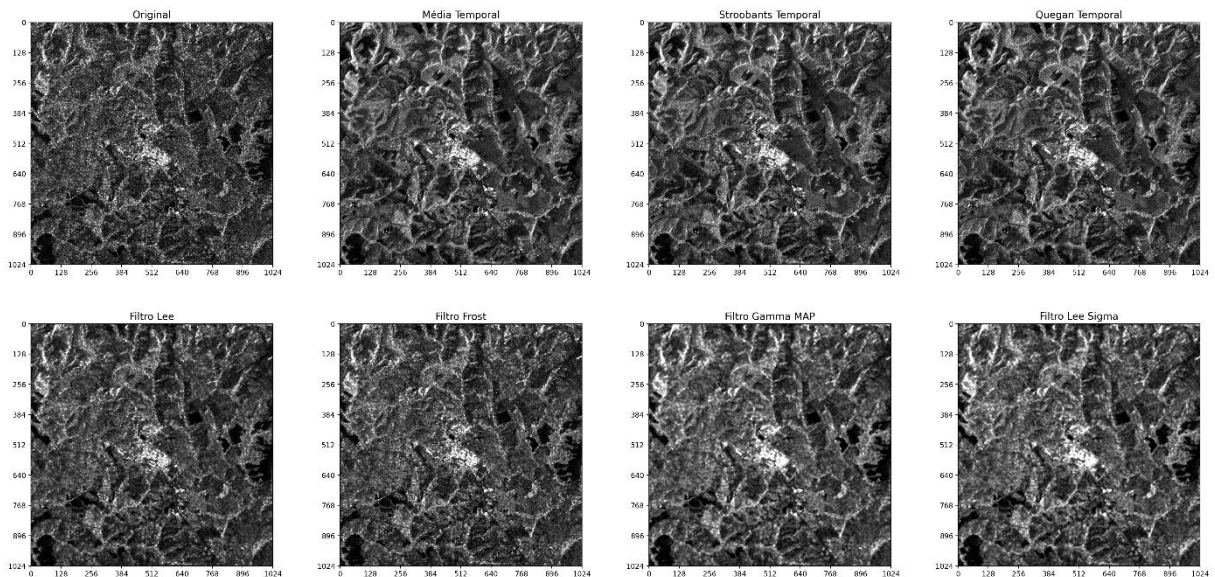


Fonte: Autoria Própria.

A análise do comportamento das métricas em função do número de imagens (Figura 10) reforça a escolha. Observa-se uma tendência de estabilização da Razão das Médias e um crescimento marginal da *GLCM Homogeneity* à medida que mais imagens são adicionadas, enquanto o *Speckle Index* apresenta uma diminuição pequena, mas progressiva. A análise da relação entre a Entropia Local e o ENL (Figura 11) é particularmente reveladora: enquanto a

Entropia Local (associada à complexidade das texturas) sofre sua maior redução ao passar de 1 para 3 imagens e depois se estabiliza em uma curva logarítmica inversa, o ENL (associada à suavidade) continua a crescer de forma quase linear. Essa dissociação clara indica que, após as 3 primeiras imagens, o processo de filtragem continua a suprimir o ruído speckle de forma eficaz sem comprometer significativamente os detalhes e as características espaciais da cena. Com base nesta análise quantitativa, o filtro de Stroobants foi selecionado como o método padrão no MASF para a geração das imagens-alvo de treinamento supervisionado. Este processo resulta em um conjunto de dados final composto por pares de imagens: as cenas individuais corregistradas como a entrada ruidosa, e a imagem filtrada por Stroobants como o alvo "limpo" correspondente.

Figura 12 – Comparação dos filtros espaciais e multitemporais aplicados ao canal VH



Fonte: Autoria Própria.

3.2.5 Cálculo de Estatísticas Globais do Conjunto de Dados

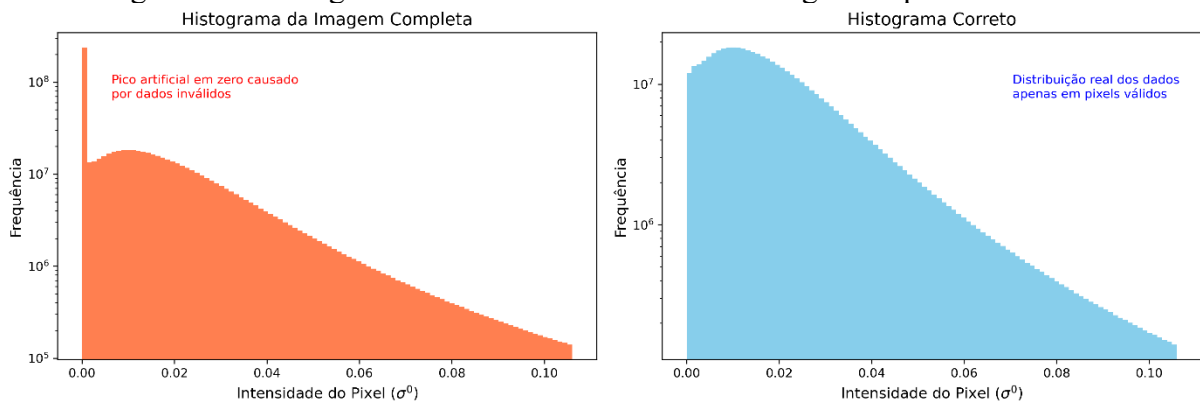
A etapa final do pré-processamento é a normalização dos dados de entrada. As redes neurais profundas são sensíveis à escala das características de entrada; grandes variações na distribuição de valores entre diferentes canais podem levar a instabilidades no treinamento e a uma convergência mais lenta. Para mitigar esse problema, adota-se a normalização Z-score (ou padronização), que reescala cada canal de entrada para que possua uma média de 0 e um desvio padrão de 1.

Antes do cálculo dessas estatísticas, aplica-se uma transformação logarítmica aos valores de σ^0 , de modo que o *speckle*, originalmente modelado como ruído multiplicativo, passe

a se comportar aproximadamente como ruído aditivo. Essa mudança de domínio torna o histograma do canal VV mais próximo de uma distribuição aproximadamente normal e, portanto, mais adequado ao uso da normalização Z-score, conforme visto na Figura 14.

O desafio reside no cálculo preciso dessas estatísticas (média e desvio padrão) para o conjunto de dados. Após a ortorretificação, as imagens SAR apresentam extensas bordas com valores nulos (pixels *no-data*), que não contêm informação radiométrica válida. Um cálculo da média e do desvio padrão sobre a imagem inteira incluiria esses milhões de pixels de valor nulo, resultando em estatísticas fortemente enviesadas que não representam a distribuição real dos dados de retroespalhamento.

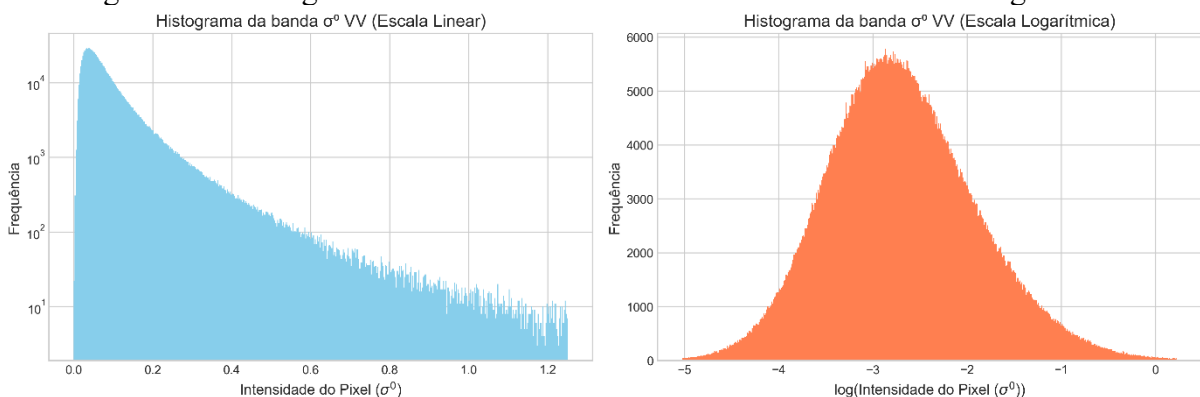
Figura 13 – Histogramas calculados sem e com a filtragem de pixels inválidos



Fonte: Autoria Própria.

Legenda: Histograma à esquerda, calculado sem filtrar valores inválidos, resultado em um pico artificial em zero, comprometendo a verdadeira distribuição dos dados. Histograma à direita, após a filtragem de valores inválidos.

Figura 14 – Histogramas do canal VV no domínio linear e no domínio logarítmico



Fonte: Autoria Própria.

Legenda: À esquerda, histograma de σ^0 no domínio linear, com assimetria típica de ruído multiplicativo; à direita, histograma após a transformação logarítmica, apresentando distribuição mais simétrica e apropriada ao cálculo de estatísticas globais e à aplicação da normalização Z-score.

Para resolver este problema, o *framework* implementa uma rotina de cálculo no script `scripts/compute_stats.py`. Este script itera sobre todas as imagens do conjunto de treinamento

e, para cada pixel, aplica um critério de validade (*e.g.*, valor $> 1e-6$) antes de incluí-lo no cálculo. Para lidar com arquivos grandes que não cabem na memória, o cálculo é realizado em blocos, acumulando a soma e a soma dos quadrados dos valores válidos, bem como a contagem total de pixels válidos por canal. Ao final do processo, a média e o desvio padrão globais para cada um dos 6 canais de entrada são calculados e salvos em **results/dataset_stats.json**, como um arquivo com o formato JSON. Este arquivo é então utilizado pelo **DataLoader** (**src/dataset_loader.py**) para normalizar cada *patch on-the-fly* antes de ser enviado ao modelo, garantindo um treinamento estável e eficiente.

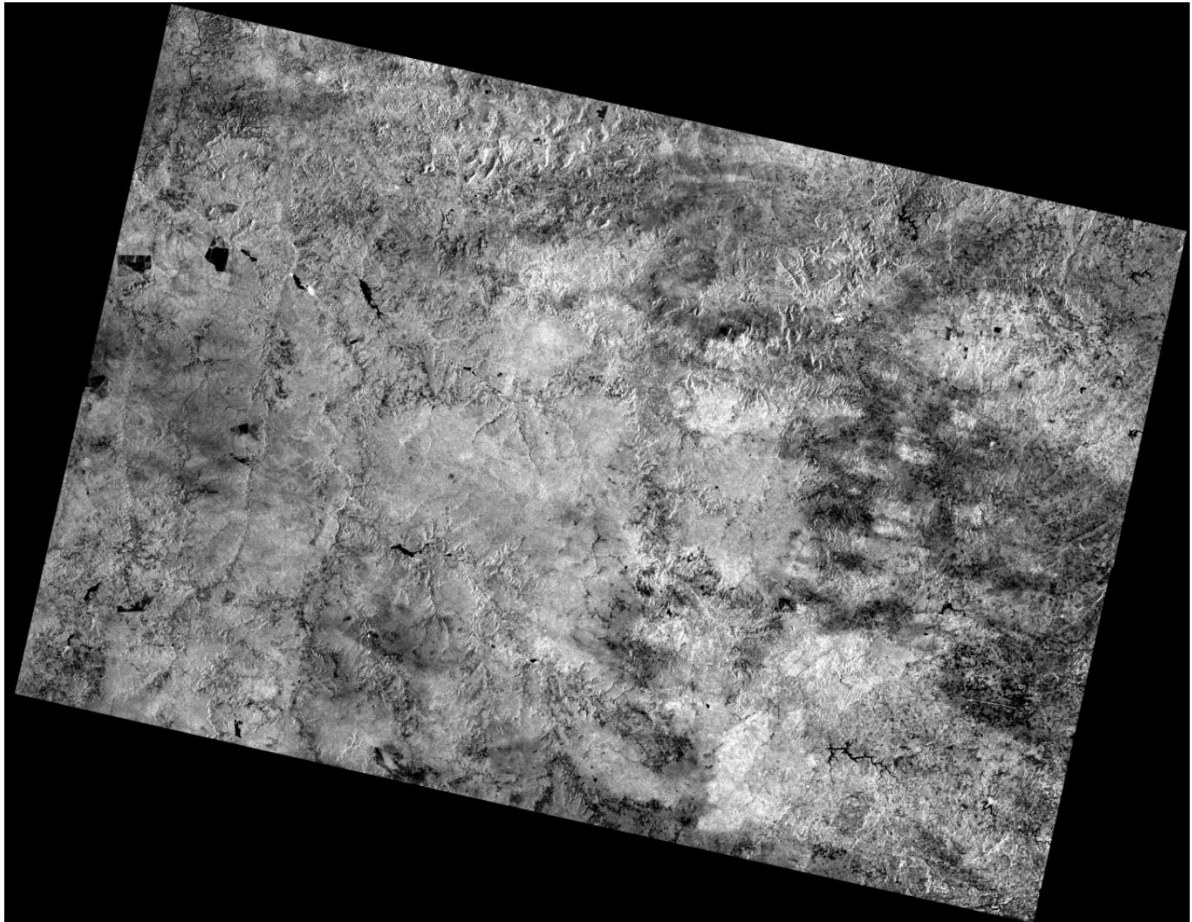
3.3 O DATALOADER CUSTOMIZADO

A ligação entre os dados pré-processados no disco e os tensores que alimentam a rede neural é o **DataLoader**, um componente que, no ecossistema PyTorch, é responsável por carregar, transformar e organizar os dados em lotes (*batches*) para o treinamento. No MASF, esta etapa é gerenciada por uma implementação customizada, a classe **S1DenoiseDataset**, definida em **src/dataset_loader.py**. Este é o componente principal do *pipeline* de preparação de dados, responsável por tarefas como a amostragem eficiente de *patches*, a aplicação *on-the-fly* de aumento de dados e a normalização.

3.3.1 Amostragem Eficiente em Imagens SAR

As imagens SAR, após a etapa de ortorretificação, raramente são retângulos perfeitos. Elas tipicamente apresentam bordas irregulares com extensas áreas de *no-data*. Uma abordagem ingênua para a extração de *patches*, como sortear aleatoriamente o canto superior esquerdo de um *patch* dentro dos limites retangulares da imagem, é extremamente ineficiente. Tal método resultaria na geração de muitos *patches* inválidos, que contêm parcela significativa ou total de pixels inválidos. Descartar esses *patches* e tentar novamente em um laço de repetição (uma técnica conhecida como amostragem por rejeição) pode retardar drasticamente o processo de treinamento, tornando o **DataLoader** ineficiente computacionalmente.

Figura 15 – Área de pixels válidos e inválidos de uma imagem SAR (Banda VH)



Fonte: Autoria Própria.

Legenda: O quadrilátero delimita a área válida, enquanto os pixels pretos representam a área inválida que deve ser filtrada.

3.3.2 Implementação da Amostragem e Cache por Localização

Para superar o desafio da amostragem em imagens com geometrias de dados válidos não retangulares, o **S1DenoiseDataset** implementa uma solução de duas etapas, projetada para ser geometricamente precisa e computacionalmente eficiente. A abordagem tradicional de simplesmente encolher um retângulo delimitador (*bounding box*) foi descartada, pois levaria a um subaproveitamento da área de dados ou ao risco de amostrar pixels inválidos devido à natureza de paralelogramo da faixa de dados SAR ortorretificada.

Primeiramente, para cada localização geográfica, o *framework* executa um pré-cálculo de metadados, implementado na função `_precompute_metadata_per_location()`. O processo ocorre da seguinte forma:

a) Geração da Nuvem de Pontos: A primeira imagem da localização é lida em baixa resolução para otimização de memória. Todos os pixels com valor maior que zero (dados

válidos) são identificados, e suas coordenadas (x, y) são coletadas, formando uma densa nuvem de pontos que representa a área de dados da cena;

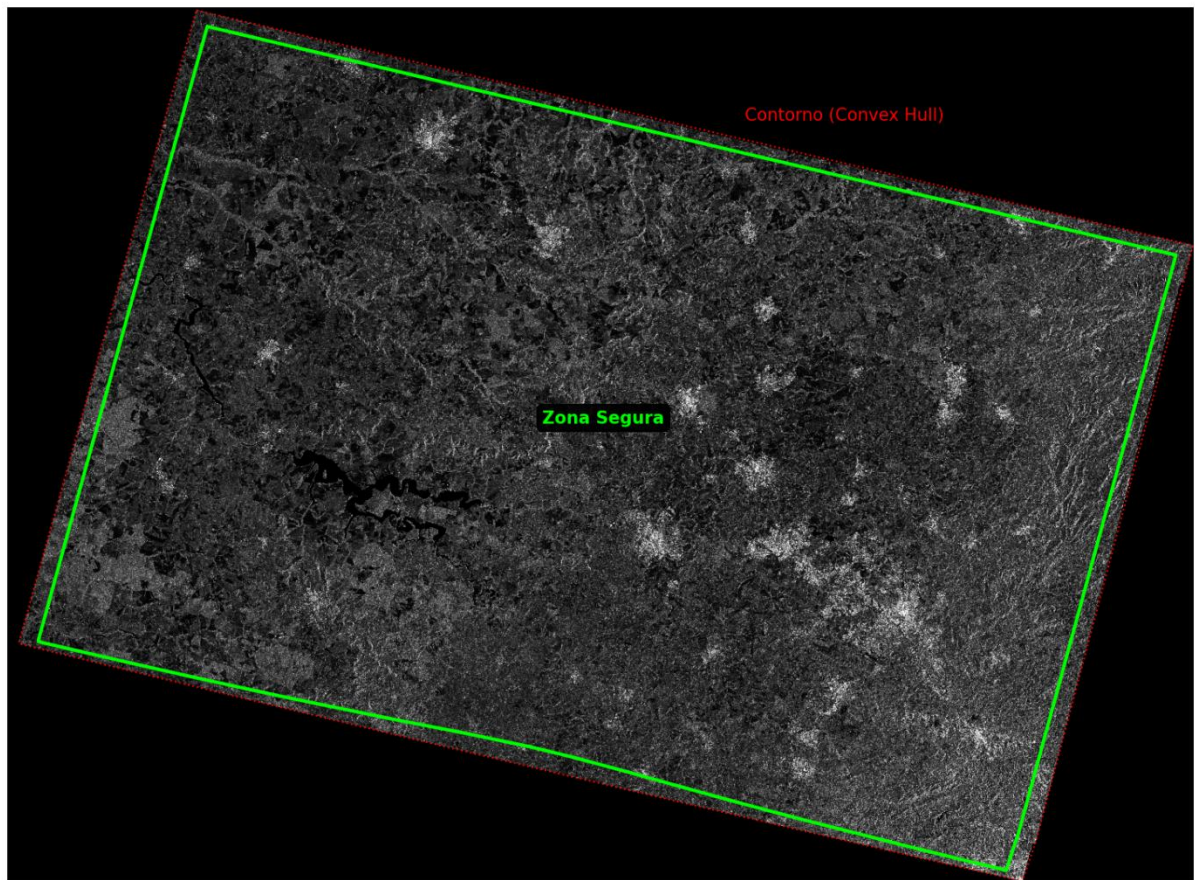
b) Determinação do Contorno Externo: Utilizando o algoritmo de Casco Convexo (*Convex Hull*) (DE BERG *et al.*, 2008) sobre a nuvem de pontos, o contorno poligonal exato que envolve todos os dados válidos é determinado. Este polígono representa fielmente o paralelogramo da faixa de dados válidos;

c) Criação da "Zona Segura": A partir do polígono do contorno externo, uma "zona segura" interna é calculada. Este processo "encolhe" o polígono para dentro por uma margem de segurança igual ao tamanho do *patch* de amostragem. O resultado é um polígono interno menor, garantindo que qualquer *patch* cuja coordenada superior esquerda seja sorteada dentro desta zona estará inteiramente contido na área de dados válidos.

E para acelerar drasticamente a inicialização do *framework* em novas execuções, os vértices que definem o polígono da "zona segura" são salvos em um arquivo de cache (.pkl) no diretório **data/04_metadata_cache/**. Como todas as imagens de uma mesma localização foram corrigidas para a mesma grade geométrica, esta zona segura é válida para todas elas. Nas próximas vezes que o treinamento for iniciado, o **DataLoader** carregará essas coordenadas poligonais diretamente do cache, eliminando a necessidade de qualquer recálculo.

Com a zona segura pré-calculada, o método **__getitem__()**, torna-se extremamente eficiente. Ele utiliza uma forma otimizada de amostragem por rejeição: sorteia coordenadas (x, y) aleatoriamente dentro do retângulo delimitador da zona segura e, em seguida, usa um teste de ponto em polígono para verificar se o ponto está, de fato, dentro dos limites do polígono. Esta verificação é computacionalmente muito barata e garante que 100% dos *patches* gerados sejam válidos, maximizando a eficiência do carregamento de dados.

Figura 16 – Área de pixels válidos determinada e a “zona segura” calculada



Fonte: Autoria Própria.

Legenda: O quadrilátero delimita a área válida, enquanto os pixels pretos representam a área inválida que deve ser filtrada.

3.3.3 Pipeline de Augmentação de Dados Implementado

Após a extração de um par de *patches* (ruidoso e alvo), mas antes da normalização, o **S1DenoiseDataset** aplica o *pipeline* de aumento de dados, cujos fundamentos teóricos foram apresentados no capítulo 2. O processo é coordenado de dentro do método `__getitem__()`:

a) **Augmentação Física**: Com uma probabilidade pré-definida, o *patch* de entrada com ruído aplica a augmentação, passando a imagem para a função `apply_random_physical_aug()` de `src/augmentations/physical_augs.py`. Esta função aplica uma composição aleatória das nove transformações físicas, alterando as propriedades radiométricas e geométricas da entrada. A aplicação dessas augmentações é seletiva para respeitar a natureza física de cada canal: perturbações que simulam ruído de sensor (*e.g.*, `add_speckle`, `add_thermal_noise`, `calibration_error`) são aplicadas apenas aos canais de retroespalhamento (VV e VH, sendo o canal da razão VV/VH recalculado), enquanto transformações geométricas (`orbital_shift`, `geometric_distortion`) são aplicadas a todos os canais para manter sua consistência espacial;

b) **Augmentação Perceptual**: Em seguida, o par de *patches* (o com ruído, possivelmente já alterado pela etapa anterior, e o alvo limpo) é passado para a função **random_flip_rot** de **src/augmentations/perceptual_augs.py**. A mesma transformação geométrica (inversão e/ou rotação) é aplicada a ambos, garantindo a consistência espacial.

Finalmente, os *patches* resultantes são normalizados utilizando as estatísticas globais pré-calculadas e convertidos em tensores PyTorch, prontos para serem agrupados em um lote pelo **DataLoader**.

3.4 ARQUITETURA DO MODELO MASF IMPLEMENTADA

A concepção da arquitetura do MASF foi guiada pelo objetivo de criar um sistema que não apenas aprendesse a mapear imagens SAR com ruído para imagens limpas, mas que o fizesse de maneira robusta, preservando detalhes finos e a estrutura radiométrica da cena. Para alcançar este objetivo, o MASF foi construído sobre a base da arquitetura U-Net, aprimorada com inovações contemporâneas em visão computacional, como *backbones* baseados em atenção, mecanismos de fusão de características inteligentes e um paradigma de aprendizado multitarefa. Esta seção detalha a estrutura e o funcionamento de cada um desses componentes, que são implementados nos scripts localizados em **src/models/**.

3.4.1 Estrutura Base U-Net e os Módulos Convolucionais

A base do MASF é a arquitetura U-Net. O caminho de codificação (*encoder*) é responsável por extrair mapas de características (*feature maps*) hierárquicos, reduzindo progressivamente a resolução espacial da entrada enquanto aumenta a sua profundidade (número de canais), capturando assim informações contextuais. O caminho de decodificação (*decoder*), por sua vez, realiza o processo inverso: ele aumenta a resolução espacial dos mapas de características de forma gradual, utilizando as informações contextuais aprendidas para reconstruir a imagem de saída detalhada.

A implementação desses caminhos é realizada por meio de blocos convolucionais modulares, definidos em **src/models/unet_block.py**. O bloco principal, **ConvBlock**, constitui tanto o *encoder* quanto o *decoder*. Ele consiste em duas camadas de convolução 2D sequenciais, cada uma com um *kernel* de 3x3 e preenchimento (*padding*) de 1 para preservar a resolução espacial. Cada camada convolucional é seguida por uma camada de normalização e uma função de ativação não-linear.

Como camada de normalização, foi adotada a **InstanceNorm2d**. Diferentemente da **BatchNorm2d**, que normaliza as ativações ao longo de todo o lote de dados, a **InstanceNorm2d** opera de forma independente em cada amostra e em cada canal. Essa característica é particularmente vantajosa em tarefas de restauração de imagem, pois a normalização remove informações de estilo específicas da instância, como contraste e brilho, forçando o modelo a focar na preservação da estrutura do conteúdo. Como função de ativação, foi utilizada a *Leaky Rectified Linear Unit* (LeakyReLU) com um coeficiente de inclinação negativo de 0,1, que ajuda a mitigar o problema do "neurônio morto" ao permitir a passagem de um pequeno gradiente para valores de entrada negativos.

Para o caminho de decodificação, o bloco **UpConvBlock** é empregado. Este módulo primeiramente aplica uma operação de *upsampling* utilizando interpolação bilinear para dobrar a altura e a largura do mapa de características. Em seguida, uma convolução 2D é aplicada para refinar as características e reduzir sua profundidade pela metade, preparando o tensor para a concatenação com as informações provenientes da *skip connection*.

3.4.2 Implementação do Bottleneck de Atenção

O gargalo (*bottleneck*) da U-Net é a região de maior compressão de informação, onde os mapas de características atingem sua menor resolução espacial e maior profundidade. É nesta camada que o modelo deve aprender a representação mais abstrata e contextual da imagem. No MASF, o *bottleneck* convolucional padrão foi substituído por arquiteturas baseadas em atenção para aprimorar a capacidade de modelar dependências de longo alcance, uma característica essencial para a remoção de ruído em texturas complexas e regiões extensas.

A primeira implementação, **SwinTransformerBlock**, baseia-se na arquitetura *Swin Transformer*. Este bloco opera sobre os mapas de características dividindo-os em janelas locais não sobrepostas. Dentro de cada janela, é aplicado um mecanismo de autoatenção multicabeça (*multi-head self-attention*), que permite que cada *pixel* (ou *patch*) pondere a importância de todos os outros *pixels* na mesma janela para construir sua nova representação. Para capturar um contexto global, blocos subsequentes aplicam um deslocamento cíclico às janelas antes de repetir o cálculo da atenção. Essa estratégia de janelas deslocadas permite que a informação flua entre janelas adjacentes, aproximando uma atenção global com complexidade computacional linear em relação ao tamanho da imagem. A implementação dessa lógica foi feita em `src/models/swin_block.py`, incluindo as camadas de normalização (**LayerNorm**) e a

rede *Multi-Layer Perceptron* (MLP) que sucedem o módulo de atenção, conforme a arquitetura original.

A segunda implementação, **MambaBlock**, utiliza um Modelo de Espaço de Estados (*State Space Model* - SSM) para capturar dependências de longo alcance. Originalmente projetado para dados sequenciais 1D, o *Mamba* foi adaptado para operar em mapas de características 2D. A implementação em `src/models/mamba_block.py` realiza essa adaptação, processando os dados de forma sequencial em múltiplas direções: os *pixels* da imagem são "achatados" em sequências ao longo da largura (da esquerda para a direita e da direita para a esquerda) e da altura (de cima para baixo e de baixo para cima). Cada uma dessas quatro sequências é processada por uma instância independente do módulo *Mamba*. Os resultados das quatro direções são então agregados para formar o mapa de características de saída. Este mecanismo de varredura multidirecional permite que o *Mamba* construa uma representação contextual global com complexidade linear, oferecendo uma alternativa computacionalmente eficiente aos *Transformers*.

3.4.3 Fusão Inteligente de Características com SkipGate

As conexões residuais (*skip connections*) são um pilar da arquitetura U-Net, pois permitem que o *decoder* acesse diretamente as características de alta resolução extraídas pelo *encoder*. Isso ajuda a preservar detalhes espaciais finos que poderiam ser perdidos no processo de compressão e expansão da rede. No entanto, a concatenação direta pode introduzir informações irrelevantes ou com ruído no *decoder*, prejudicando a reconstrução.

Para mitigar esse problema, o *framework* MASF implementa um mecanismo de atenção customizado chamado *SkipGate*, definido em `src/models/unet_block.py`. Em vez de simplesmente concatenar os tensores do *encoder* e do *decoder*, o *SkipGate* atua como um portão de controle, aprendendo a ponderar a importância das características provenientes do *encoder* antes da fusão.

O mecanismo funciona da seguinte forma:

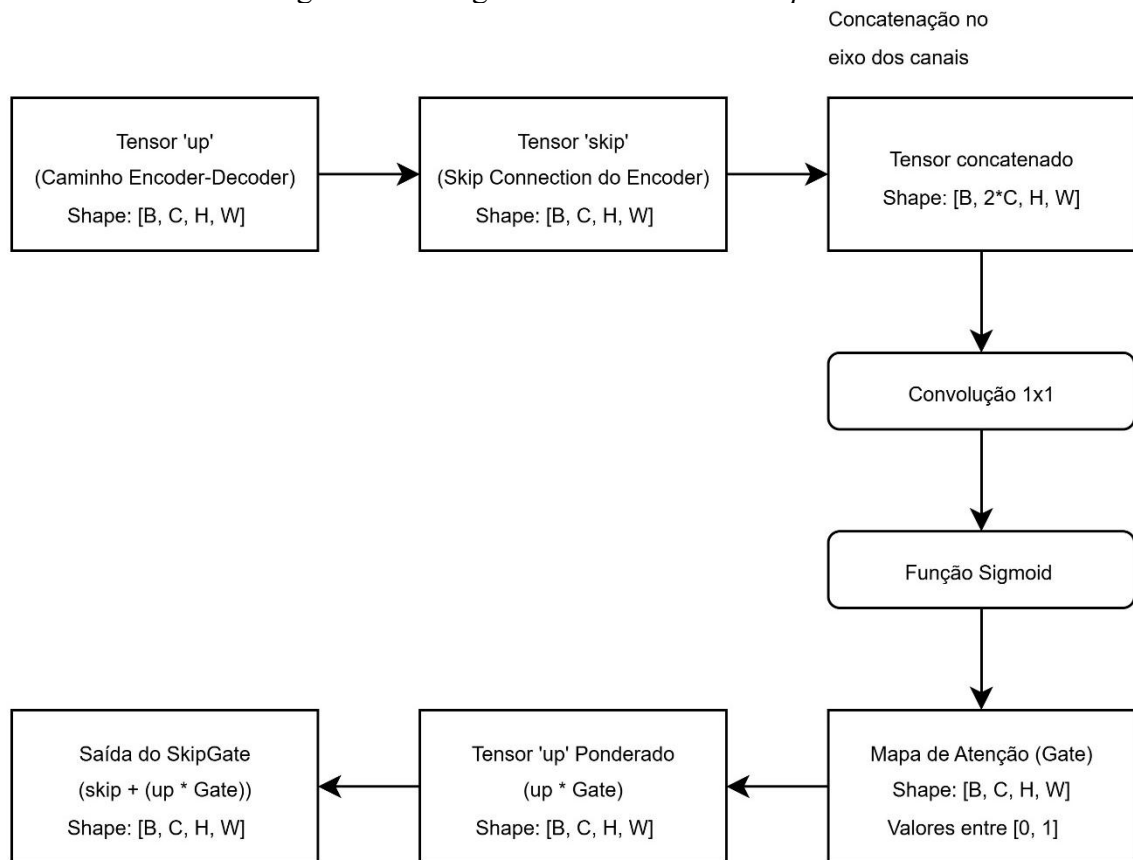
- a) O tensor d vindo do caminho de decodificação (após o *upsampling*) e o tensor e vindo do caminho de codificação (*skip connection*) são concatenados ao longo do eixo dos canais;
- b) Este tensor concatenado é passado por uma convolução 1×1 , que reduz sua profundidade de volta à profundidade original e . Essa convolução atua como uma camada linear que aprende a combinar as informações de ambos os caminhos;

c) O resultado é então passado por uma função de ativação Sigmoid, que mapeia cada valor para o intervalo $[0, 1]$. O tensor resultante, g , funciona como um mapa de atenção ou "portão" (*gate*), onde valores próximos de 1 indicam características importantes e valores próximos de 0 indicam características a serem suprimidas;

d) Este mapa de atenção g é multiplicado elemento a elemento pelo tensor original da *skip connection* e, efetivamente filtrando-o e retraindo apenas as informações mais relevantes;

e) Finalmente, o tensor filtrado $e_{filtrado}$ é concatenado com o tensor d do *decoder*, e o resultado é passado para o próximo bloco convolucional do caminho de decodificação.

Figura 17 – Diagrama do mecanismo *SkipGate*



Fonte: Autoria Própria.

Essa abordagem de fusão permite que o modelo aprenda dinamicamente quais detalhes espaciais de cada nível de resolução são mais úteis para a tarefa de supressão de ruído, resultando em uma reconstrução mais fiel e com menos artefatos.

3.4.4 Implementação da Arquitetura Multitarefa

Para explorar a possibilidade de regularização implícita e o aprendizado de representações mais ricas, o MASF foi estendido para suportar o aprendizado multitarefa. A implementação, na classe **MultiTaskUNet** em `src/models/multi_task_model.py`, utiliza uma abordagem de compartilhamento de parâmetro, onde a maior parte da rede é compartilhada entre as tarefas, e apenas as camadas finais são específicas para cada uma.

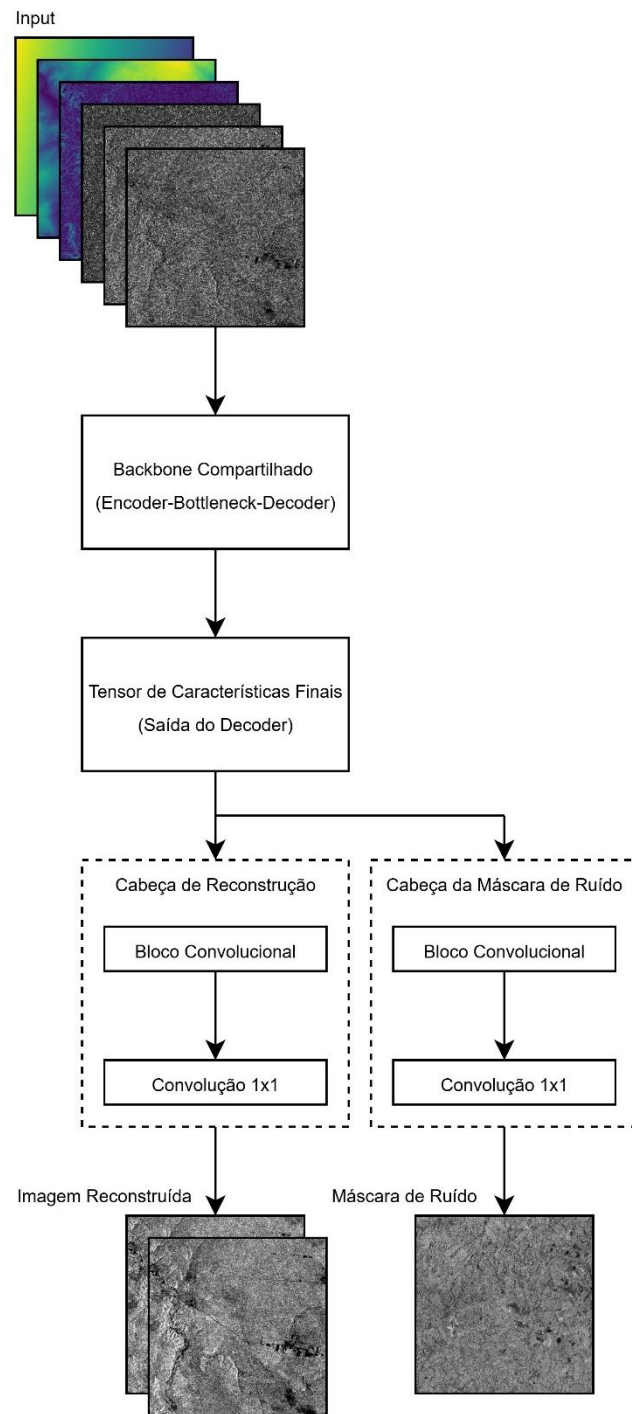
A classe **MultiTaskUNet** funciona como um invólucro (*wrapper*) que recebe um modelo de *backbone* (seja o **MASF_Swin** ou o **MASF_Mamba**) como sua base. O *encoder*, o *bottleneck* e o *decoder* deste *backbone* são compartilhados, aprendendo um conjunto de características genéricas úteis para todas as tarefas.

Na saída do *decoder* do *backbone*, em vez de uma única camada de predição, duas "cabeças" (*heads*) de saída independentes são anexadas. Cada cabeça é uma pequena rede neural, implementada como uma ou duas camadas convolucionais, que mapeia o mapa de características compartilhado para a saída de uma tarefa específica. Conforme definido na configuração do projeto e implementado no código, as duas tarefas são:

a) Cabeça de Reconstrução: A tarefa principal, responsável por gerar a imagem limpa a partir da entrada ruidosa. Sua saída é comparada à imagem-alvo multitemporal através da perda perceptual composta;

b) Cabeça de Predição da Máscara de Ruído: Uma tarefa auxiliar, responsável por prever o ruído residual (a diferença entre a entrada ruidosa e a saída limpa). Sua saída é comparada à máscara de ruído *ground truth* (calculada como $I_{ruidosa} - I_{alvo}$) através de uma perda mais simples, como a L1.

Figura 18 – Diagrama da arquitetura multitarefa



Fonte: Autoria Própria.

A hipótese do aprendizado multitarefa (*multi-task learn* – MTL) é que forçar o modelo a aprender simultaneamente a estrutura do sinal (reconstrução) e a estrutura do ruído (predição da máscara) atua como uma forte forma de regularização indutiva. O aprendizado da tarefa auxiliar de predição do ruído pode ajudar o modelo a desenvolver representações internas mais robustas e a dissociar de forma mais eficaz o sinal da imagem do componente de ruído, melhorando o desempenho na tarefa principal.

3.5 ESTRATÉGIA DE TREINAMENTO E OTIMIZAÇÃO

A estratégia de treinamento é um componente crucial que define como o MASF aprende a suprimir o ruído speckle. Em vez de uma abordagem única, o *framework* adota um *pipeline* de treinamento sofisticado em dois estágios, projetado para maximizar tanto a fidelidade perceptual quanto a acurácia estatística dos resultados.

O primeiro estágio consiste em um treinamento supervisionado clássico, no qual o modelo aprende a mapear imagens SAR com ruído para seus respectivos alvos limpos, gerados por filtragem multitemporal. O objetivo desta fase é ensinar ao modelo a estrutura geral da cena e a natureza da supressão de ruído.

O segundo estágio, opcional, é um refinamento não supervisionado e informado por física. Nesta fase, o modelo pré-treinado é ajustado utilizando apenas as imagens com ruído, sem acesso aos alvos limpos. A otimização é guiada por uma função de perda que penaliza desvios das propriedades estatísticas conhecidas do ruído speckle. Essa abordagem visa corrigir sutis imperfeições e garantir que o ruído residual na imagem filtrada seja estatisticamente consistente com o modelo teórico do speckle, melhorando a robustez e a generalização do modelo.

Para garantir que os dois estágios operem com máxima eficiência, o *framework* integra um processo de otimização de hiperparâmetros sistemática utilizando a biblioteca Optuna, permitindo a descoberta de configurações ótimas para a arquitetura, o otimizador e as funções de perda. As seções a seguir detalham a implementação de cada uma dessas etapas.

3.5.1 Implementação do Treinamento Supervisionado

O treinamento supervisionado é a base do aprendizado no MASF. Seu objetivo é ensinar ao modelo uma transformação complexa que remove o ruído speckle enquanto preserva ao máximo as características radiométricas e texturais da cena. Esta etapa é coordenada pelo script **scripts/train.py**, que gerencia todo o ciclo de vida do treinamento.

O processo inicia com o carregamento da configuração centralizada do projeto, a construção dos **DataLoaders** de treinamento e validação e a instanciação do modelo e do otimizador. O otimizador escolhido foi o AdamW (LOSHCHILOV; HUTTER, 2017), uma variante do otimizador Adam que desacopla o mecanismo de decaimento de peso (*weight decay*) da atualização do gradiente. Essa modificação melhora a regularização e frequentemente leva a uma melhor generalização do modelo.

O núcleo do treinamento supervisionado reside na função de perda, que quantifica a diferença entre a predição do modelo e o alvo limpo. Em vez de depender de uma única métrica, como o erro quadrático médio (L2), que tende a produzir resultados suaves e borrar detalhes finos, o MASF emprega uma função de perda perceptual composta, implementada em **src/losses/perceptual_losses.py**. Esta função é uma soma ponderada de múltiplas perdas, cada uma focada em um aspecto diferente da qualidade da imagem.

A perda total neste estágio é definida pela Equação (13), onde os pesos λ_i são hiperparâmetros que controlam a contribuição de cada componente e são ajustados durante a fase de otimização de hiperparâmetros.

O laço de treinamento principal, contido em **scripts/train.py**, itera sobre as épocas. Em cada época, o script processa todos os lotes de dados do **DataLoader** de treinamento, executando o *forward pass*, calculando a perda composta, realizando a retropropagação do erro (*backpropagation*) e atualizando os pesos do modelo com o otimizador. Ao final de cada época, o modelo é avaliado no conjunto de validação para monitorar o *overfitting*. A lógica de *checkpointing*, implementada em **src/utlis/checkpoint_utils.py**, salva o estado do modelo periodicamente, mantendo tanto o último *checkpoint* quanto o melhor modelo com base na perda de validação.

3.5.2 Implementação do Refinamento Não Supervisionado

Embora o treinamento supervisionado seja eficaz, ele pode levar o modelo a aprender "atalhos" que resultam em imagens visualmente plausíveis, mas que não respeitam as propriedades físicas de imagens SAR. O ruído residual pode não seguir a distribuição estatística esperada do speckle, um fenômeno conhecido como *over-smoothing* em áreas homogêneas ou a introdução de artefatos correlacionados. Ademais, o modelo, por definição, estará limitado à qualidade do alvo multitemporal.

Para resolver essas limitações, o *framework* implementa um estágio de refinamento não supervisionado, inspirado em Redes Neurais Informadas por Física (RAISSI; PERDIKARIS; KARNIADAKIS, 2019). Esta etapa, gerenciada pelo script **scripts/refine_unsupervised.py**, não utiliza os alvos limpos. Em vez disso, otimiza um modelo pré-treinado usando uma função de perda física composta que opera sobre o ruído residual, $n_{res} = I_{log} - \hat{R}_{log}$, onde I_{log} é a imagem de entrada ruidosa em domínio logarítmico e $\hat{R}_{log}$ é a predição do modelo.

A função de perda física, implementada em `src/losses/physical_losses.py`, é uma soma ponderada de componentes que forçam o ruído residual a se conformar com as propriedades estatísticas teóricas do speckle em domínio logarítmico, definida pela Equação (18).

O script `scripts/refine_unsupervised.py` carrega o melhor modelo salvo do estágio supervisionado e continua seu treinamento por um número adicional de épocas. No entanto, o laço de treinamento utiliza apenas as imagens com ruído do conjunto de dados e otimiza os pesos do modelo com base na $\mathcal{L}_{física}$. Tipicamente, uma taxa de aprendizado muito menor é usada nesta fase para garantir que o modelo refine seu comportamento sem "esquecer" o conhecimento estrutural aprendido na fase supervisionada.

Quadro 2 – Comparativo entre os estágios de treinamento

Característica	Treinamento Supervisionado	Refinamento Não-Supervisionado
Script Principal	train.py	refine_unsupervised.py
Dados de Entrada	Imagem com ruído, Alvo limpo	Apenas imagem com ruído
Função de Perda	$\mathcal{L}_{perceptual}$	$\mathcal{L}_{física}$
Domínio da Perda	Comparação (Predição vs. Alvo)	Análise do Ruído Residual
Objetivo Principal	Aprender a estrutura da cena e a remoção geral do ruído.	Corrigir a estatística do ruído residual para maior fidelidade física.

Fonte: Autoria Própria.

3.5.3 Otimização de Hiperparâmetros com Optuna

O desempenho de um modelo de *deep learning* é altamente sensível à escolha de seus hiperparâmetros, como a taxa de aprendizado, a profundidade da rede ou os pesos das funções de perda. A seleção manual desses valores é um processo demorado e que dificilmente encontra os valores ótimos globais. Para abordar este desafio de forma sistemática, o MASF integra um *pipeline* de otimização de hiperparâmetros automatizado usando a biblioteca Optuna (AKIBA *et al.*, 2019).

O script `scripts/hyperparam_search.py` gerencia esse processo. Ele define uma função objetivo que encapsula todo o *pipeline* de treinamento e avaliação. Esta função recebe um

conjunto de hiperparâmetros sugeridos por uma triagem do Optuna, modifica temporariamente a configuração do projeto com esses valores, executa o script de treinamento **train.py** e, opcionalmente, o de refinamento **refine_unsupervised.py**, e retorna uma métrica de desempenho (tipicamente a perda de validação final) que o Optuna tentará minimizar.

O espaço de busca, ou seja, os hiperparâmetros a serem otimizados e seus respectivos intervalos, é definido na seção **hyperparameter_search** do arquivo de configuração. A busca é dividida em dois estágios independentes para refletir a estratégia de treinamento do *framework*:

a) Busca de Treinamento: Foca nos hiperparâmetros mais influentes da arquitetura e do treinamento principal. Isso inclui a escolha do otimizador (Adam vs. AdamW), a taxa de aprendizado, o decaimento de peso, a dimensão dos *embeddings* do modelo, a profundidade dos blocos de atenção e os pesos da perda perceptual;

b) Busca de Refinamento: Uma vez que os melhores hiperparâmetros da fase supervisionada são encontrados e fixados na configuração principal, uma segunda busca, menor e mais focada, é executada para otimizar a taxa de aprendizado do refinamento e os pesos da perda física.

O Optuna utiliza algoritmos de amostragem eficientes, como o TPE (*Tree-structured Parzen Estimator*), para explorar o espaço de busca de forma inteligente, concentrando-se em regiões mais promissoras com base nos resultados de *trials* anteriores. Todos os resultados da busca são armazenados em um banco de dados SQLite, o que permite a persistência, a retomada de buscas interrompidas e a análise posterior dos resultados com o script **scripts/visualize_hyperparam_search.py**, que gera gráficos da importância dos parâmetros e do histórico de otimização.

3.6 PROCEDIMENTO EXPERIMENTAL

A validação de um novo *framework* de supressão de ruído exige um procedimento experimental que seja, ao mesmo tempo, rigoroso, reprodutível e abrangente. Para avaliar de forma conclusiva o desempenho do MASF, foi estabelecida uma metodologia de avaliação projetada para responder a três questões:

a) Desempenho Comparativo: O modelo MASF proposto supera os métodos de filtragem clássicos e estabelecidos na literatura?

b) Contribuição dos Componentes: Qual é o impacto real e quantificável das inovações dos componentes do *framework*, como a estratégia de treinamento em dois estágios, as augmentações de dados e os mecanismos de atenção?

c) Reprodutibilidade: O ambiente experimental está suficientemente bem definido para que os resultados possam ser verificados e replicados por outros pesquisadores?

Para abordar essas questões, o procedimento foi estruturado em torno de uma configuração de ambiente controlada, um conjunto diversificado de métricas de avaliação, uma comparação direta com métodos de referência (*baselines*) e uma série de estudos de ablação sistemáticos.

3.6.1 Configuração do Ambiente Computacional

Para garantir que todos os resultados apresentados neste trabalho possam ser replicados de forma idêntica, o ambiente computacional foi rigorosamente controlado e documentado. A gestão do ambiente foi realizada através do sistema de pacotes Miniconda 3, com todas as dependências e suas versões exatas especificadas no arquivo **environment.yaml**, localizado na raiz do projeto.

O ecossistema de software é baseado em bibliotecas de código aberto, com destaque para:

- a) Python (v3.11): Linguagem de programação base para todo o desenvolvimento;
- b) PyTorch (v2.5.0) com suporte a CUDA (v11.8): Utilizado como o *framework* de *deep learning* para a implementação, treinamento e inferência de todos os modelos. O treinamento foi acelerado por hardware, utilizando uma Unidade de Processamento Gráfico (GPU) NVIDIA, um requisito para a execução dos experimentos;
- c) Bibliotecas Geoespaciais (GDAL v3.9.0, Rasterio): Empregadas para a manipulação de dados geoespaciais no formato GeoTIFF, garantindo a integridade das informações de georreferenciamento durante todo o processo;
- d) Ecossistema Científico (NumPy v2.0, Scikit-Image, SciPy): Utilizadas para computação numérica de alto desempenho, sendo a base para a implementação das métricas de avaliação e dos filtros clássicos;
- e) Optuna: Biblioteca para otimização de hiperparâmetros, utilizada para encontrar as configurações ideais do modelo, conforme detalhado na seção 3.5.3.

A definição explícita deste ambiente garante que as diferenças nos resultados possam ser atribuídas às variações experimentais controladas, e não a inconsistências no software.

3.6.2 Métricas de Avaliação Adotadas

A qualidade de uma imagem SAR filtrada é uma propriedade complexa que não pode ser capturada por uma única métrica. Um filtro eficaz deve, simultaneamente, reduzir o ruído, preservar bordas e feições lineares, manter a textura de áreas heterogêneas e não distorcer a radiometria da cena. Para realizar uma avaliação ampla, o MASF emprega um conjunto complementar de métricas, implementadas no módulo `src/utlis/metrics.py` e definidas na configuração do projeto, agrupadas em cinco domínios de avaliação distintos:

a) Métricas de Fidelidade e Similaridade Estrutural: Este domínio avalia o quão fiel a imagem filtrada é em relação ao alvo limpo, sendo composto por métricas exclusivamente referenciais. O *Peak Signal-to-Noise Ratio* (PSNR) quantifica o erro pixel a pixel em uma escala logarítmica. Embora seja um padrão na literatura, é conhecido por nem sempre se correlacionar bem com a qualidade visual percebida. Para suprir essa lacuna, o *Structural Similarity Index* (SSIM) (WANG *et al.*, 2004) oferece uma medida mais alinhada à percepção humana, avaliando a similaridade com base na luminância, contraste e estrutura;

b) Métricas de Supressão de Ruído: O principal objetivo de um filtro de ruído speckle é suavizar o ruído em regiões homogêneas. O *Equivalent Number of Looks* (ENL) é a métrica para quantificar essa suavidade. Um ENL mais alto indica uma supressão de speckle mais eficaz. O ENL pode ser usada tanto de forma referencial (calculada sobre o alvo para definir a suavidade ideal) quanto não-referencial (calculada sobre a imagem filtrada para medir a suavidade alcançada). De forma complementar, o *Speckle Index* (SI), uma métrica não-referencial estima a quantidade de ruído residual na imagem filtrada;

c) Métricas de Preservação de Bordas: Um bom filtro deve remover o ruído sem borrar feições importantes. Para avaliar isso, são utilizadas métricas referenciais como o *Edge Preservation Degree* (EPD), que mede a correlação dos gradientes de alta frequência, e o *Pratt's Figure of Merit* (FOM), que avalia a precisão dos contornos detectados. De forma não-referencial, o *Edge Preservation Index* (EPI) compara os gradientes da imagem filtrada com os da imagem ruidosa original para quantificar se a estrutura principal da cena foi mantida;

d) Métricas de Consistência Radiométrica: Estas métricas avaliam se o filtro preservou as propriedades estatísticas e o brilho geral da imagem. De forma referencial, a Distância de Kolmogorov-Smirnov (K-S) e a Divergência de Kullback-Leibler (KL) comparam as distribuições de probabilidade (histogramas) da imagem filtrada e do alvo. De forma não-referencial, o *Ratio of Means* (RoM) verifica se o brilho médio da imagem foi preservado em relação à imagem com ruído original, com um valor ideal próximo de 1,0;

e) Métricas de Análise de Textura: Este conjunto de métricas avalia como as características texturais locais são afetadas pela filtragem. A Entropia de Shannon, usada de forma referencial, descreve a complexidade ou "informação" da imagem alvo. As métricas não-referenciais, baseadas na Matriz de Coocorrência de Níveis de Cinza (GLCM), como *GLCM Homogeneity* e *GLCM Contrast*, quantificam a suavidade e a variação da textura local na imagem filtrada. A Entropia Local também mede a complexidade da textura local. Espera-se que um bom filtro aumente a homogeneidade e diminua o contraste e a entropia local em áreas que não contêm feições.

A utilização conjunta dessas métricas, abrangendo os cinco domínios, permite uma análise detalhada e robusta das vantagens e desvantagens de cada método de filtragem.

Quadro 3 – Domínio das métricas de avaliação

Domínio da Avaliação	Métrica Referencial	Métrica Não-Referencial
Fidelidade e Estrutura	SSIM, PSNR	-
Supressão de Ruído	ENL, Índice de Speckle (<i>Speckle Index</i>)	ENL, Índice de Speckle (<i>Speckle Index</i>)
Preservação de Bordas	EPD, FOM	EPI
Consistência Radiométrica	Kullback-Leibler, Kolmogorov-Smirnov	Razão das Médias (<i>Ratio of Means</i>)
Análise de Textura	Entropia de Shannon	GLCM, Entropia Local

Fonte: Autoria Própria.

3.6.3 Métodos de Referência (Baselines) para Comparação

Para contextualizar o desempenho do MASF, é fundamental compará-lo com métodos de referência (*baselines*) em filtragem de ruído speckle. Foram selecionados quatro filtros clássicos, amplamente citados na literatura e utilizados em aplicações comerciais e acadêmicas. Estes filtros, que operam com base em estatísticas locais da imagem, foram implementados no módulo `src/baseline/classical_filters.py` para garantir uma comparação justa e controlada.

Os *baselines* selecionados, cujos parâmetros como o tamanho da janela são definidos no arquivo de configuração, são:

- a) Filtro de Lee (LEE, 1986): Um filtro adaptativo que utiliza a média e a variância locais;
- b) Filtro de Frost (FROST *et al.*, 1982): Similar ao Lee, mas com um núcleo de convolução com pesos que decaem exponencialmente;
- c) Filtro Lee Sigma: Uma variação do filtro de Lee que melhora a preservação de bordas através de um critério de exclusão de *outliers*;
- d) Filtro Gamma MAP (LOPES; TOUZI; NEZRY, 1990): Assume que o sinal segue uma distribuição Gama, o que modela o speckle com alta precisão estatística.

O script `scripts/compare_baselines.py` executa essa avaliação comparativa. Ele aplica o modelo MASF treinado e todos os filtros *baseline* ao mesmo conjunto de dados de teste, calcula o conjunto completo de métricas e gera os resultados quantitativos e qualitativos que são apresentados no Capítulo 4.

3.6.4 Estrutura dos Estudos de Ablação

O MASF é composto por múltiplos componentes inovadores. Para isolar e validar a contribuição de cada um, foram projetados estudos de ablação. Nesta metodologia, versões simplificadas do *framework* são treinadas e avaliadas, onde um componente específico é removido ou substituído por uma alternativa mais simples. A diferença de desempenho entre a versão com ablação e o modelo completo revela a importância do componente em questão.

O *framework* foi concebido para automatizar este processo. A seção **ablation_studies** do arquivo de configuração permite a definição declarativa de cada experimento. O script `scripts/run_ablation_studies.py` lê essa configuração, executa o *pipeline* completo de treinamento e avaliação para cada cenário de forma sequencial e, ao final, agrega todos os resultados em uma tabela comparativa.

Os seguintes estudos de ablação foram definidos para investigar hipóteses específicas:

- a) **Baseline**: O modelo MASF completo, que serve como referência de desempenho;
- b) **no_physical_aug**: Remove as augmentações físicas. Hipótese: A simulação de variações físicas no sensor torna o modelo mais robusto a dados do mundo real;
- c) **l1_only_loss**: Substitui a perda perceptual composta por uma perda L1 simples; Hipótese: A combinação de perdas estruturais (SSIM) e de gradiente produz resultados visualmente superiores.

d) **no_skip_gates**: Substitui os SkipGate por uma concatenação direta. Hipótese: O mecanismo de atenção nas conexões residuais melhora a fusão de características e a preservação de detalhes;

e) **no_refinement**: Desativa o estágio de refinamento não supervisionado. Hipótese: O refinamento informado por física corrige imperfeições estatísticas, melhorando as métricas de qualidade e a fidelidade do ruído residual.

Os resultados desses experimentos, apresentados no Capítulo 4, fornecem uma validação empírica para as escolhas de *design* arquitetural e de treinamento feitas no desenvolvimento do MASF.

4 RESULTADOS E DISCUSSÃO

Este capítulo apresenta os resultados empíricos obtidos através do procedimento experimental, validando a eficácia do MASF. A exposição dos resultados é sequencial: inicia-se com a otimização de hiperparâmetros para definir o modelo de melhor desempenho; prossegue com uma análise comparativa, quantitativa e qualitativa, deste modelo contra os métodos de referência; aprofunda-se na interpretabilidade do modelo através da visualização de seus mecanismos de atenção; e conclui com a discussão dos estudos de ablação, que quantificam a contribuição de cada componente arquitetural e metodológico do *framework*.

4.1 OTIMIZAÇÃO DE HIPERPARÂMETROS E SELEÇÃO DO MODELO

A etapa inicial foi a otimização de hiperparâmetros do estágio de treinamento supervisionado, utilizando o *framework* Optuna, para três de cinco configurações de modelo avaliadas neste trabalho: Swin com perda L1, Swin com perda $\mathcal{L}_{perceptual}$ e Mamba com perda $\mathcal{L}_{perceptual}$. Para cada uma das configurações, conforme descrito na Seção 3.5.3, foram executadas 60 tentativas de minimizar a função de perda no conjunto de validação.

Em cada tentativa, o Optuna amostrava um conjunto de hiperparâmetros dentro de intervalos previamente estabelecidos visando suas otimizações, conforme a Tabela 3. Os demais parâmetros se encontram no APÊNDICE B – parâmetros do modelo.

Tabela 3 – Hiperparâmetros otimizados para diferentes configurações do modelo

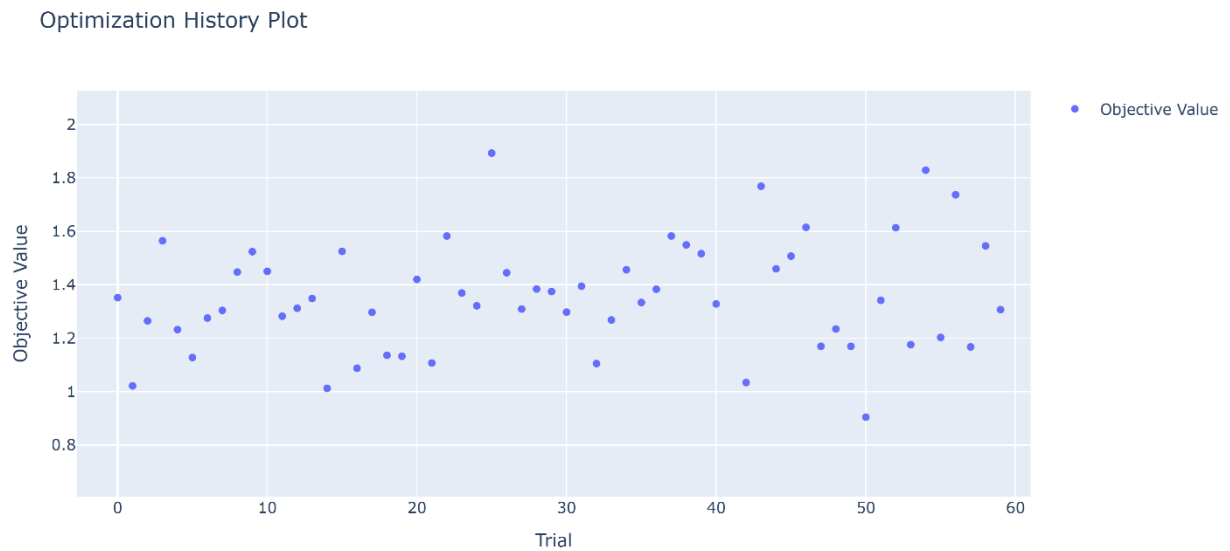
Modelo	Parâmetros	Intervalos	Valores otimizados
MASF-Swin (L1)	optimizer	Adam ou AdamW	Adam
	swin_heads	2, 3, 4, 6 ou 12	4
	swin_depth	[2, 6]	3
	learning_rate	[1e-5, 1e-3]	0,00029168438660425404
	weight_decay	[1e-6, 1e-2]	0,000014130544093928408
MASF-Swin ($\mathcal{L}_{perceptual}$)	optimizer	Adam ou AdamW	Adam
	swin_heads	2, 3, 4, 6 ou 12	4
	swin_depth	[2, 6]	3
	learning_rate	[1e-5, 1e-3]	0,00037805283863190283
	weight_decay	[1e-6, 1e-2]	0,002790527067529427
MASF-Mamba ($\mathcal{L}_{perceptual}$)	optimizer	Adam ou AdamW	AdamW
	mamba_d_state	8, 16 ou 32	16
	mamba_depth	[2, 6]	3
	learning_rate	[1e-5, 1e-3]	0,0009684762914968829
	weight_decay	[1e-6, 1e-2]	0,001676563343956272

Fonte: Autoria Própria.

As imagens abaixo apresentam, para a configuração de Swin com perda $\mathcal{L}_{perceptual}$, o comportamento típico observado no processo de otimização de hiperparâmetros.

Observa-se na Figura 19 o histórico da otimização, que mostra a evolução da função de perda em função do número de tentativas. Cada ponto no gráfico corresponde ao valor da perda de validação obtido para uma combinação específica de hiperparâmetros. Diferente de uma otimização bayesiana tradicional, onde após uma fase inicial de exploração o algoritmo passa a concentrar as amostragens, para essa tarefa observou-se uma dispersão contínua nas tentativas de otimização.

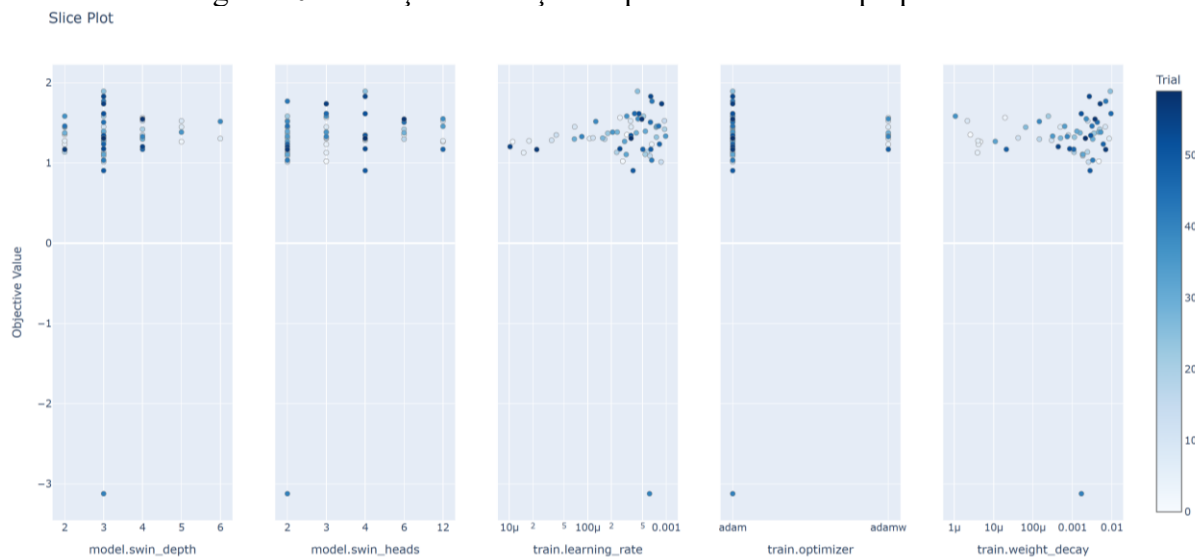
Figura 19 – Histórico de otimização dos hiperparâmetros com Optuna



Fonte: Autoria Própria.

A Figura 20 apresenta os *slice plots* gerados pelo Optuna, que representam a relação entre o valor da função de perda e cada hiperparâmetro listado na Tabela 3, conforme as combinações avaliadas durante a busca. Esses gráficos permitem analisar, de forma qualitativa, a sensibilidade do desempenho do modelo a variações individuais dos hiperparâmetros e identificar faixas de valores mais estáveis, nas quais pequenas perturbações não produzem degradação significativa na perda.

Figura 20 – Relação da função de perda com cada hiperparâmetro



Fonte: Autoria Própria.

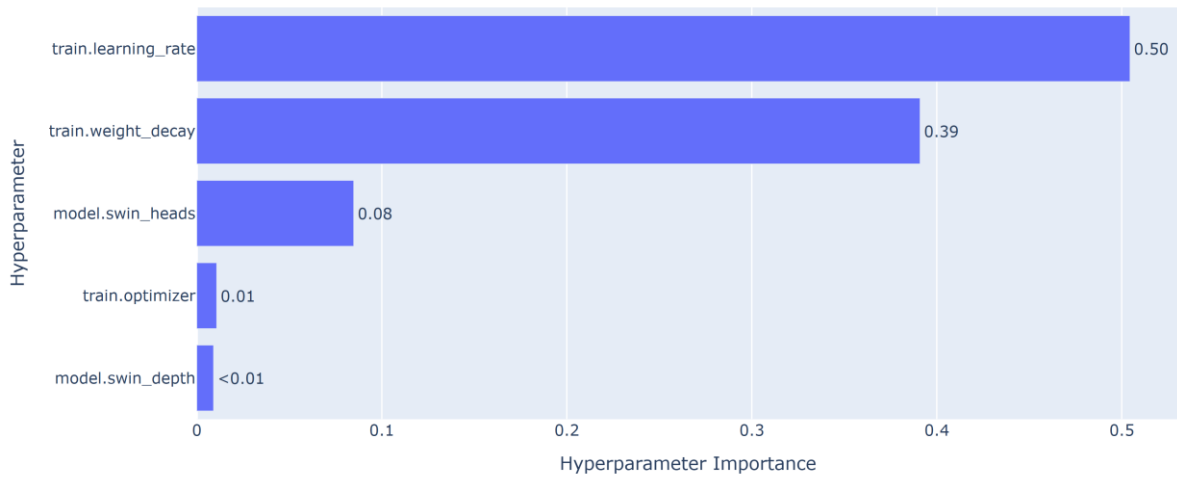
De modo geral, observa-se que:

- a) A taxa de aprendizado (**learning_rate**) apresenta uma faixa intermediária na qual a perda de validação tende a assumir valores menores, enquanto taxas muito baixas resultam em treinamento lento e taxas muito altas induzem instabilidade no treinamento;
- b) O fator de decaimento (**weight_decay**) também apresenta um comportamento característico, em que valores muito pequenos levam a instabilidade, e valores muito elevados prejudicam a capacidade de representação do modelo;
- c) Um número de cabeças do *backbone* (**swin_heads**) e sua profundidade (**swin_depth**) mais elevados induz estabilidade ao custo de taxas mais lentas de aprendizado;
- d) Para essa configuração, o otimizador (**optimizer**) AdamW é mais estável, mas Adam atinge perdas de validação potencialmente menores.

Essas curvas não são superfícies de resposta suaves e contínuas, mas sim amostras discretas obtidas a partir do conjunto de tentativas. Ainda assim, elas ajudam a justificar as faixas de valores adotadas como “regiões seguras” para o treinamento final do modelo.

A Figura 21 apresenta a importância relativa de cada hiperparâmetro para a função de perda, também estimada pelo Optuna. Verifica-se que os parâmetros diretamente ligados ao processo de otimização apresentam maior influência sobre o desempenho global, enquanto os parâmetros estruturais exercem papel complementar.

Figura 21 – Importância relativa dos hiperparâmetros analisados
Hyperparameter Importances



Fonte: Autoria Própria.

Para cada uma das três configurações de modelo, foi selecionada a combinação de hiperparâmetros que minimizou a perda de validação, e os respectivos valores ótimos estão consolidados na Tabela 3. Essas configurações foram então fixadas para:

- Treinar o modelo nas condições definitivas adotadas no estudo;
- Gerar as predições analisadas nas seções subsequentes deste capítulo;
- Comparar o *framework* proposto com os métodos de referência.

Dessa forma, o processo de otimização de hiperparâmetros garante que as comparações apresentadas neste capítulo sejam realizadas a partir de configurações representativas e próximas de um ótimo local para cada cenário, reduzindo o viés decorrente de escolhas arbitrárias de parâmetros e aumentando a reprodutibilidade do estudo.

Para o cenário de refinamento, o único hiperparâmetro relevante, a taxa de aprendizado, foi fixada em um valor muito menor que a do modelo principal para reduzir as chances de esquecimento, enquanto o decaimento de peso foi zerado. Já para o treinamento multitarefa, utilizou-se os hiperparâmetros do MASF-Swin com perda perceptual, mas com a divisão de pesos em 90% para a tarefa de reconstrução e 10% para geração da máscara de ruído.

4.2 ANÁLISE QUANTITATIVA COMPARATIVA

Com os hiperparâmetros otimizados para cada configuração de modelo, conforme descrito na Seção 4.1, foram obtidos os resultados quantitativos na etapa de treino do *framework*

proposto, comparando-se o desempenho do MASF com os filtros e com as variantes do próprio modelo. As métricas foram calculadas a partir dos *patches* de validação do canal VV.

Os resultados da comparação entre o MASF e os filtros clássicos são apresentados na Tabela 4. De forma geral, observa-se que o *framework* proposto alcança valores de ENL significativamente superiores aos dos métodos clássicos, indicando uma supressão de *speckle* mais agressiva em regiões homogêneas. Em contrapartida, o MASF apresenta valores de PSNR e SSIM em níveis inferiores aos dos filtros clássicos. Há, portanto, compromisso entre redução de ruído e fidelidade radiométrica: quanto maior a homogeneização das áreas estáveis, maior a penalização em termos de erro médio e similaridade estrutural.

Tabela 4 – Média das métricas na análise quantitativa

Modelo	PSNR	SSIM	ENL	EPD	FOM	KS	KL	Shannon
Lee	29,647	0,704	1,276	0,534	0,286	0,108	0,090	4,306
Frost	28,261	0,594	1,311	0,343	0,246	0,119	0,075	4,613
Quegan	30,722	0,707	1,587	0,470	0,181	0,105	0,050	4,995
Lee Sigma	31,302	0,72	2,182	0,459	0,062	0,113	0,099	5,543
Gamma MAP	31,227	0,722	1,690	0,490	0,126	0,109	0,068	5,364
MASF-Swin (L1)	24,499	0,584	6,565	0,452	0,036	0,468	7,094	5,412
MASF-Swin ($\mathcal{L}_{perceptual}$)	26,162	0,623	4,333	0,480	0,091	0,388	4,036	4,789
MASF-Mamba ($\mathcal{L}_{perceptual}$)	24,343	0,570	6,110	0,468	0,062	0,502	6,483	5,220
MASF-Swin Multitask	23,880	0,583	4,063	0,419	0,087	0,465	5,666	4,513
MASF-Swin Refinado	26,320	0,648	2,652	0,459	0,078	0,335	2,259	4,510

Fonte: Autoria Própria.

Os filtros clássicos ainda apresentam vantagens em alguns indicadores específicos. O filtro de Lee, por exemplo, obtém os maiores valores de FOM e EPD, refletindo uma preservação superior de bordas finas em comparação com o MASF. Por outro lado, quando as métricas são analisadas em conjunto, nota-se que as variantes analisadas, em especial o MASF-Swin ($\mathcal{L}_{perceptual}$) e o MASF-Swin Refinado, alcançam ENL relativamente superiores aos dos filtros clássicos, mantendo a Entropia de Shannon em níveis próximos aos métodos clássicos.

Isso significa que a cena permanece rica em textura e variabilidade espacial, ao mesmo tempo em que o ruído é fortemente reduzido. No caso do MASF-Swin (L1), porém, a combinação de ENL muito alto e alta entropia está associada à presença de artefatos e realces artificiais, enquanto no MASF-Swin ($\mathcal{L}_{perceptual}$) o aumento de ENL ocorre com textura mais natural e coerente, como corroborado pela análise qualitativa. Assim, embora PSNR e SSIM sejam ligeiramente inferiores aos dos filtros clássicos, o balanço entre ENL, entropia e inspeção visual indica um ganho real de qualidade de textura espacial para as variantes mais bem calibradas do MASF.

Ao avaliar as variantes do MASF na Tabela 4, verifica-se o seguinte comportamento:

- a) O MASF-Swin (L1) apresenta os maiores valores de ENL entre todos os modelos, indicando forte supressão de ruído, porém à custa de PSNR e SSIM mais baixos e de FOM extremamente reduzido, o que evidencia suavização excessiva e perda de estrutura fina;
- b) O MASF-Swin ($\mathcal{L}_{perceptual}$) melhora PSNR e SSIM em relação ao modelo com L1 e reduz moderadamente o ENL, obtendo um compromisso mais equilibrado entre suavização e preservação de detalhes, com ganhos em FOM e redução de KL em relação ao L1;
- c) A variante MASF-Mamba ($\mathcal{L}_{perceptual}$) apresenta desempenho semelhante ao Swin em termos médios, com ENL elevado e PSNR/SSIM mais baixos, não se destacando de forma consistente em relação ao Swin para o conjunto de métricas adotado;
- d) O MASF-Swin Multitask não se sobressai em nenhuma métrica específica, mantendo PSNR e SSIM relativamente baixos e valores de ENL intermediários;
- e) O MASF-Swin Refinado alcança os maiores valores de PSNR e SSIM entre as variantes do MASF, ao mesmo tempo em que reduz o ENL em relação às demais configurações do *framework*, aproximando-se dos níveis de homogeneidade dos filtros clássicos, mas ainda mantendo ENL superior. Em termos estatísticos, apresenta também os menores valores de KS e KL dentro das variantes do MASF, embora ainda superiores aos dos filtros clássicos, o que indica uma melhora relativa na aderência à distribuição de referência quando comparado às demais variantes.

Tabela 5 – Desvio padrão das métricas na análise quantitativa

Modelo	PSNR	SSIM	ENL	EPD	FOM	KS	KL	Shannon
Lee	7,509	0,286	0,768	0,267	0,307	0,035	0,049	1,096
Frost	7,634	0,334	0,931	0,21	0,268	0,069	0,123	1,066
Quegan	8,053	0,295	1,103	0,251	0,224	0,041	0,037	1,095
Lee Sigma	7,052	0,303	1,451	0,246	0,11	0,035	0,061	1,098
Gamma MAP	7,65	0,29	1,173	0,26	0,174	0,068	0,083	1,047
MASF-Swin (L1)	10,47	0,269	5,855	0,201	0,129	0,321	9,994	1,001
MASF-Swin ($\mathcal{L}_{perceptual}$)	9,257	0,258	3,908	0,195	0,152	0,304	8,079	0,991
MASF-Mamba ($\mathcal{L}_{perceptual}$)	11,187	0,286	5,933	0,229	0,111	0,309	8,577	1,026
MASF-Swin Multitask	10,298	0,274	4,221	0,21	0,17	0,325	9,124	1,051
MASF-Swin Refinado	10,045	0,253	2,187	0,191	0,142	0,258	5,068	1,039

Fonte: Autoria Própria.

Na Tabela 5, que apresenta o desvio padrão das métricas, observa-se que as variantes do MASF exibem dispersão significativamente maior em ENL, KS e, principalmente, KL do que os filtros clássicos. Isso indica que o desempenho estatístico do framework proposto é mais heterogêneo entre as diferentes cenas, com casos em que a aderência à distribuição teórica se aproxima do comportamento desejado e outros em que há divergências mais acentuadas. Uma possível medida de aprimoramento futuro consiste em reduzir essa variabilidade, buscando estratégias de treinamento e regularização que tornem o comportamento do modelo mais consistente em diferentes condições de cena.

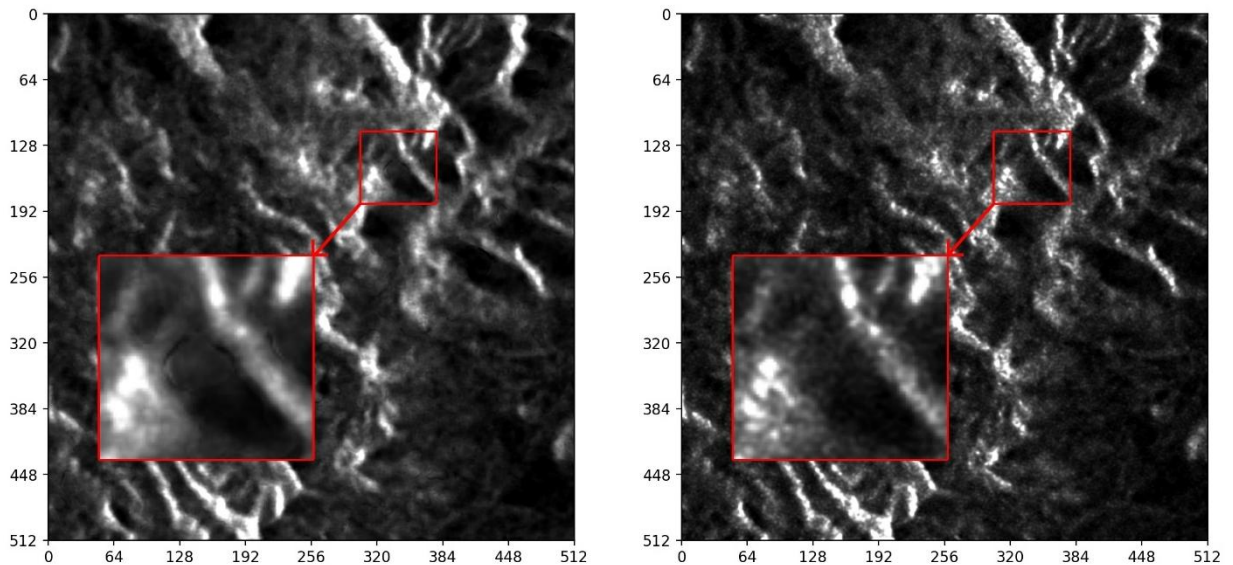
Em síntese, os resultados quantitativos mostram que o MASF alcança ganhos substanciais em homogeneização do *speckle*, ao custo de PSNR/SSIM inferiores e de uma aderência estatística menos fiel que a dos filtros clássicos. Dentro das variantes propostas, o MASF-Swin com perda perceptual oferece um compromisso melhor entre suavização e preservação de detalhes, enquanto o MASF-Swin Refinado melhora a coerência estatística relativa às demais variante, ainda que sem superar os filtros clássicos nesse aspecto.

4.3 ANÁLISE QUALITATIVA COMPARATIVA

A avaliação quantitativa é complementada por uma análise qualitativa dos modelos de supressão de ruído, permitindo observar o impacto visual de cada método sobre texturas, bordas e artefatos residuais.

Na Figura 22, observa-se, no retângulo realçado, a presença de artefatos residuais após o treinamento com a perda L1 (à esquerda) que são suprimidos quando se utiliza a perda $\mathcal{L}_{perceptual}$ (à direita). A supressão desses artefatos está de acordo com os ganhos de PSNR e SSIM, bem como a redução da Entropia de Shannon apresentados na Tabela 4, indicando que a perda perceptual contribui para uma reconstrução visualmente superior, com melhor preservação da estrutura local.

Figura 22 – Artefatos suprimidos no modelo treinado com perda perceptual

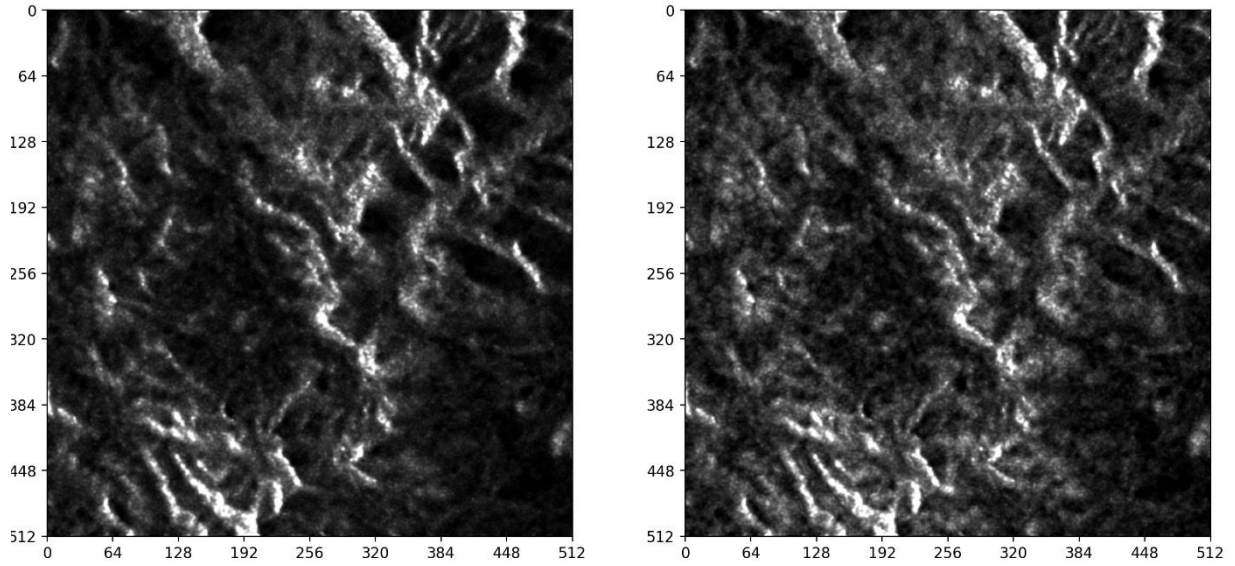


Fonte: Autoria Própria.

Legenda: Comparação entre MASF-Swin treinado com L1 (à esquerda) e MASF-Swin com perda perceptual (à direita) na supressão de artefatos.

A Figura 23 compara a saída gerada pelo modelo treinado com a perda $\mathcal{L}_{perceptual}$ (à esquerda) e o modelo treinado com a perda $\mathcal{L}_{perceptual}$ e refinado com a perda $\mathcal{L}_{física}$ (à direita). Nessa comparação, observa-se um ganho de textura espacial no modelo refinado, associado à redução dos valores médios de KS e KL em relação à distribuição teórica do ruído *speckle*. Essa melhoria estatística é obtida sem degradação relevante em PSNR e SSIM, mas ao custo de uma redução do ENL, isto é, da reintrodução parcial de ruído, que se manifesta como um leve aumento de granularidade nas regiões homogêneas.

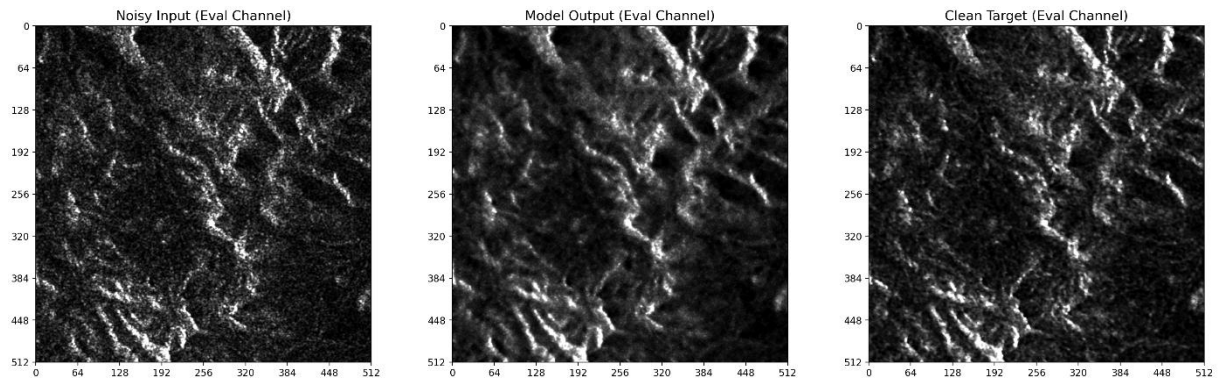
Figura 23 – Imagem restaurada com modelo treinado vs. modelo refinado



Fonte: Autoria Própria.

A Figura 24 compara a imagem restaurada com o MASF-Swin treinado com perda $\mathcal{L}_{perceptual}$ ao alvo multitemporal de referência, evidenciando que, em muitas regiões, a textura e o contraste da imagem restaurada se aproximam do comportamento médio observado ao longo da série temporal. Figura 25 compara a imagem restaurada por este mesmo modelo com os demais métodos de supressão de ruído.

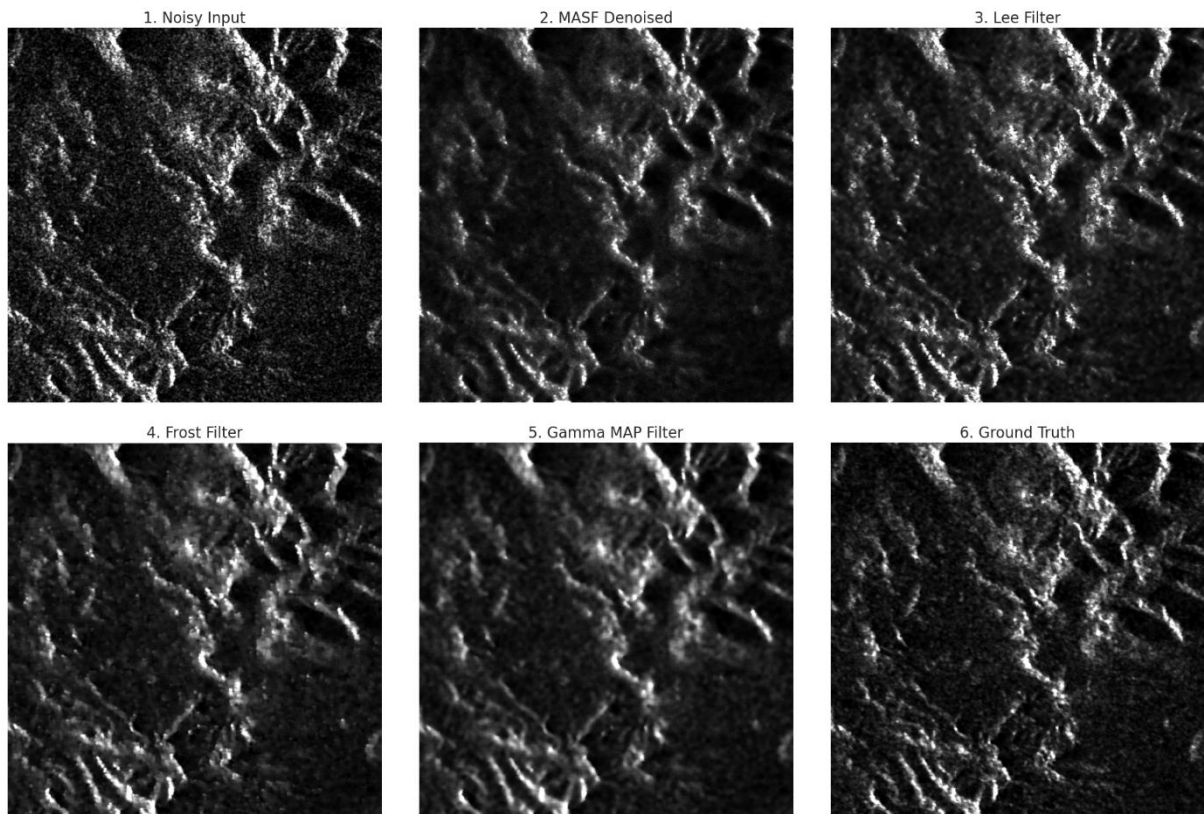
Figura 24 – Imagem restaurada com modelo treinado vs. modelo refinado



Fonte: Autoria Própria.

Já a Figura 25 apresenta uma comparação qualitativa entre a imagem original com ruído, a saída do MASF-Swin com perda perceptual, os filtros clássicos (Lee, Frost e Gamma MAP) e o alvo multitemporal (Stroobants) para um mesmo recorte do canal VV.

Figura 25 – Comparação qualitativa dos métodos de supressão de ruído



Fonte: Autoria Própria.

Legenda: Comparação qualitativa entre a imagem original, o MASF (Swin com perda perceptual), os filtros clássicos (Lee, Frost e Gamma MAP) e o alvo multitemporal de referência (Stroobants) para o canal VV.

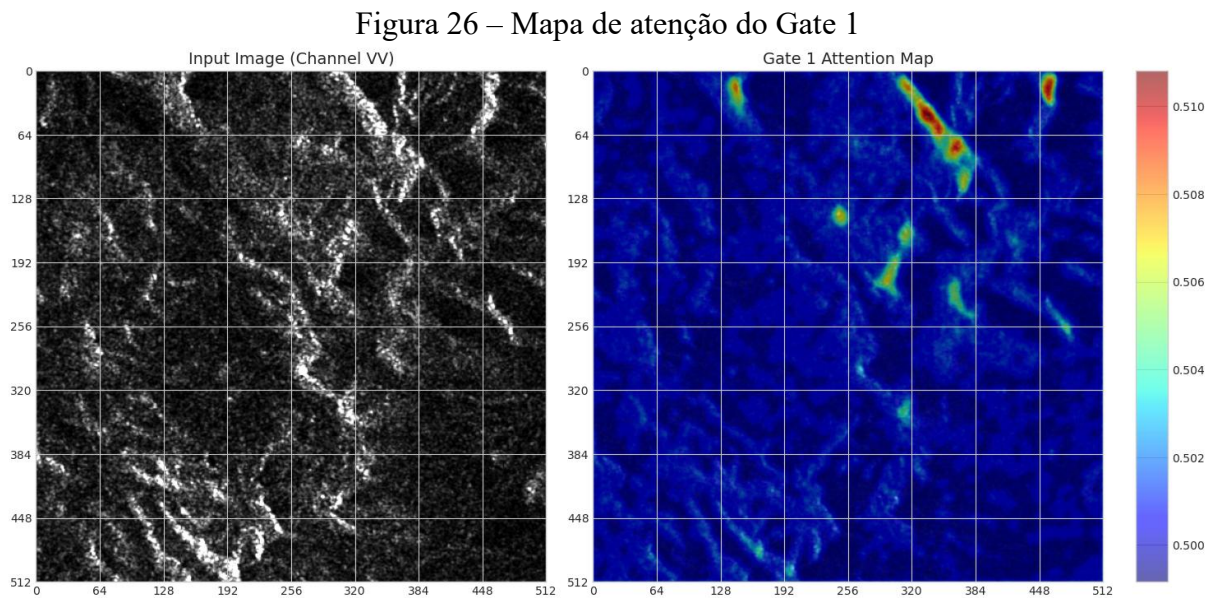
Na imagem com ruído, nota-se um padrão de *speckle* intenso em toda a cena, comprometendo detalhes finos e dificultando a interpretação visual. O MASF produz uma imagem visivelmente mais suave em regiões homogêneas, com textura de fundo regularizada e sem introdução perceptível de artefatos estruturais, apresentando em diversas áreas uma qualidade comparável, e por vezes superior, à do próprio alvo multitemporal. As bordas principais, em transições bruscas de retroespalhamento, são as mais bem preservadas, embora algumas bordas muito finas apresentem leve atenuação, em acordo com os resultados de FOM.

O filtro de Lee apresenta imagens com menor nível de ruído que a entrada, porém com aspecto ainda granular, sobretudo em áreas de maior intensidade do sinal. Por outro lado, preserva bem bordas, o que ajuda a explicar seus bons resultados em métricas de borda. O filtro Gamma MAP tende a produzir uma imagem mais suave que Lee em regiões homogêneas, mas com evidente borramento de detalhes, enquanto Frost apresenta uma suavização intermediária, equilibrando redução de ruído e preservação de textura, mas sem atingir o mesmo grau de homogeneização obtido pelo MASF.

De modo geral, a percepção visual é de que o MASF-Swin treinado com a perda $\mathcal{L}_{perceptual}$ produz imagens com melhor balanço entre suavização e preservação de estrutura entre as variantes analisadas, justificando seu uso como configuração principal nas comparações qualitativas. Para tarefas onde a fidelidade estatística da distribuição de intensidades é o requisito principal, entretanto, os filtros clássicos permanecem mais adequados, em função de seus melhores indicadores de KS e KL.

4.4 VISUALIZAÇÃO E INTERPRETAÇÃO DOS MAPAS DE ATENÇÃO

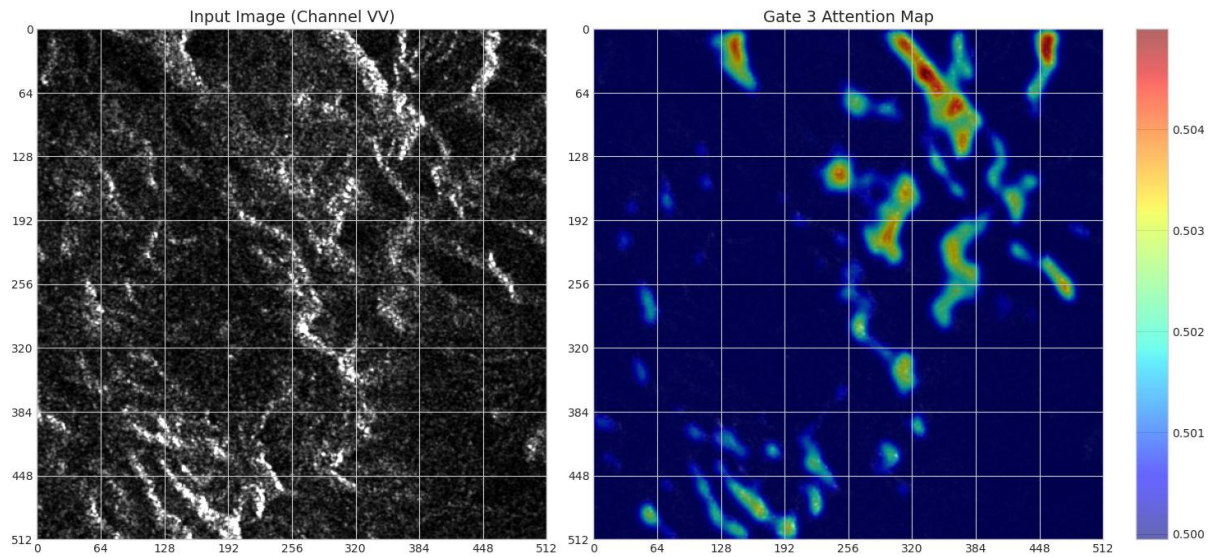
Para compreender o comportamento do modelo em suas camadas mais profundas, é necessário avaliar os mapas produzidos pelos mecanismos de atenção. A Figura 26 apresenta, lado a lado, a imagem de entrada no canal VV e o mapa de atenção do Gate 1.



Fonte: Autoria Própria.

Nota-se, na Figura 26, que o mecanismo de atenção ainda é pouco sensível às estruturas espaciais da cena, respondendo de forma difusa e uniforme, em grande parte devido ao próprio ruído, embora já capte com maior intensidade as regiões com mais ruído. Já a Figura 27 apresenta o mapa de atenção do Gate 3, em uma camada mais profunda e de menor resolução da rede, onde é possível observar que o modelo passa a destacar com maior clareza as diferenças nas estruturas espaciais, bem como regiões de menor retroespalhamento e características geomorfológicas mais marcantes.

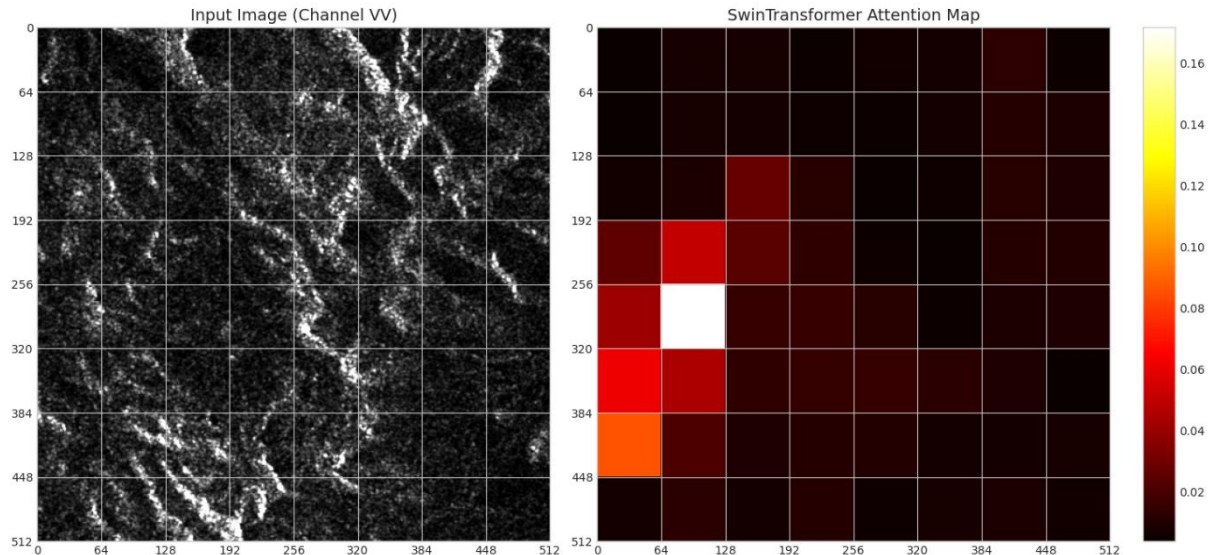
Figura 27 – Mapa de atenção do Gate 3



Fonte: Autoria Própria.

A Figura 28 apresenta o mapa de atenção do Swin no *bottleneck* da rede. Observa-se que os blocos de atenção são ativados principalmente em regiões de maior complexidade estrutural, independentemente da intensidade absoluta do sinal retroespalhado nessas áreas, evidenciando que o contexto global é um fator importante nesse estágio da arquitetura.

Figura 28 – Mapa de atenção do Swin

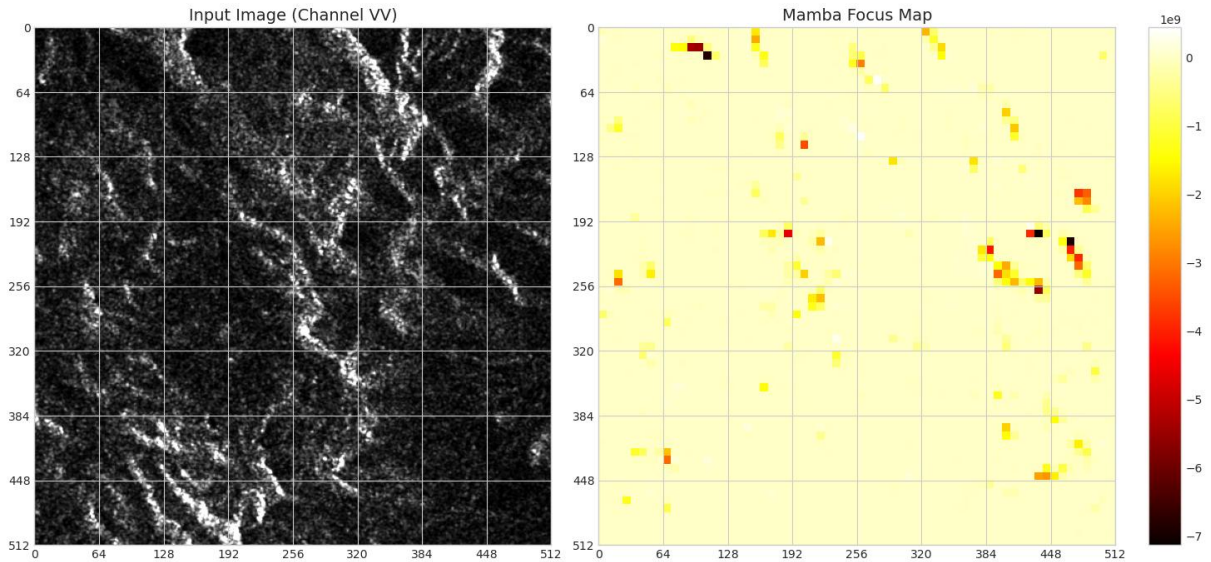


Fonte: Autoria Própria.

Por fim, a Figura 29 apresenta o mapa de atenção do Mamba no *bottleneck*. Nesse caso, observa-se um padrão menos estruturado espacialmente, o que é compatível com a natureza unidimensional do modelo de espaço de estados, originalmente concebido para sequências 1D.

Em imagens 2D SAR, essa característica limita a capacidade do Mamba de realçar seletivamente detalhes espaciais finos, quando comparado ao Swin.

Figura 29 – Mapa de atenção do Mamba



Fonte: Autoria Própria.

Essas visualizações permitem analisar como o modelo distribui sua atenção espacial ao longo da cena, ou seja, em quais regiões tende a confiar mais nos detalhes vindos das conexões SkipGate ou do *backbone* Swin. Essa perspectiva é particularmente relevante em imagens SAR, nas quais a distinção entre ruído e textura nem sempre é trivial, e um modelo poderia, em princípio, aprender estratégias de filtragem pouco amparadas pela física do retroespalhamento. A análise dos mapas de atenção sugere que, nas camadas mais profundas, o *framework* passa a concentrar seus recursos de modelagem em feições estruturalmente relevantes.

4.4.1 Análise Qualitativa dos Mapas de Atenção

No Gate 1 do SkipGate, o mapa de atenção apresenta valores pouco contrastados, com variação suave em torno de um valor e poucos padrões espaciais definidos. Visualmente, o mapa se assemelha a um campo quase homogêneo, com pequenas flutuações. Esse comportamento sugere que, nas primeiras camadas da rede, o mecanismo de *gating* atua de forma relativamente uniforme sobre as características provenientes do *encoder*, funcionando mais como uma normalização suave da informação do que como um seletor espacial.

No Gate 3, por outro lado, o mapa de atenção torna-se mais organizado. As regiões de alta atenção tendem a se concentrar ao redor de feições geomorfológicas marcantes e em áreas

de textura heterogênea, enquanto áreas de maior retroespalhamento apresentam valores de atenção mais baixos. Em termos qualitativos, isso indica que os *gates* mais profundos aprendem a suprimir contribuições dominadas por ruído em regiões pouco informativas (áreas de maior retroespalhamento, onde o ruído é mais intenso), ao mesmo tempo em que preservam e realçam detalhes em zonas críticas para a reconstrução estrutural.

O mapa de atenção associado ao Swin Transformer apresenta uma granularidade mais grosseira, refletindo a natureza baseada em janelas do mecanismo de atenção local. Ainda assim, é possível identificar um gradiente entre diferentes blocos espaciais: regiões próximas aos contornos mais proeminentes da imagem recebem maior peso relativo, enquanto blocos relacionados à área de fundo uniforme aparecem com atenção reduzida. Isso sugere que o *backbone* contribui principalmente para capturar dependências de maior escala, reforçando o contraste entre regiões de estruturas distintas, enquanto o SkipGate atua de maneira mais fina na modulação dos detalhes locais.

4.4.2 Análise da Correlação entre Atenção e Textura da Imagem

A interpretação qualitativa dos mapas de atenção foi complementada por uma análise quantitativa da correlação entre os valores de atenção do SkipGate e uma medida de textura local calculada a partir do canal VV. A Tabela 6 apresenta os coeficientes de correlação de Pearson entre atenção e textura para cada *gate* e para o *backbone* em diferentes variantes do MASF.

Tabela 6 – Análise quantitativa dos mapas de atenção

Modelo	Gate 1	Gate 2	Gate 3	Backbone
MASF-Swin (L1)	-0,199	-0,086	0,237	0,211
MASF-Swin ($\mathcal{L}_{perceptual}$)	0,345	0,401	0,415	0,422
MASF-Mamba ($\mathcal{L}_{perceptual}$)	-0,226	-0,128	0,267	-0,066
MASF-Swin Multitask	0,06	0,042	0,18	-0,182
MASF-Swin Refinado	0,261	0,387	0,416	0,416

Fonte: Autoria Própria.

Os resultados mostram que o comportamento da correlação entre atenção e textura varia entre as variantes do modelo. Para as configurações mais bem-sucedidas, em particular o MASF-Swin ($\mathcal{L}_{perceptual}$) e o MASF-Swin Refinado, os coeficientes de correlação são positivos e de magnitude moderada ($\approx 0,35-0,42$) em todos os *gates* e no *backbone*. Isso indica que, nesses casos, o modelo tende a atribuir maior atenção a regiões com textura mais complexa, reforçando a interpretação de que os *gates* mais profundos concentram seus recursos em feições estruturalmente relevantes.

Já no MASF-Swin (L1) e no MASF-Mamba ($\mathcal{L}_{perceptual}$), observa-se correlação fraca ou negativa nos *gates* iniciais e, no caso do Mamba, também no *backbone*. Esse comportamento sugere que, nessas variantes, o mecanismo de atenção não está alinhado às variações de textura local, o que é coerente com a presença de artefatos no modelo treinado com L1 e com as limitações espaciais do Mamba em dados 2D. A variante Multitask apresenta correlações fracas em todos os *gates* e correlação negativa no *backbone*, indicando uma relação difusa entre atenção e textura.

Esses resultados quantitativos corroboram a análise qualitativa:

- a) Nas variantes mais bem calibradas (Swin com perda perceptual e Swin refinado), os *gates* mais profundos e o *backbone* Swin tendem a concentrar atenção em regiões com texturas ricas, suprimindo contribuições em áreas de fundo uniforme;
- b) Em configurações menos estáveis, a relação entre atenção e textura é mais fraca ou invertida.

Em conjunto com a atuação do *backbone* Swin, responsável pelo contexto de maior escala, o SkipGate se comporta como um mecanismo de atenção local (modulação de detalhes finos), contribuindo para que o MASF realize uma supressão de *speckle* mais compatível com a estrutura física da cena do que a apenas os critérios locais.

4.5 DISCUSSÃO DOS ESTUDOS DE ABLAÇÃO

Os resultados quantitativos e qualitativos apresentados nas seções anteriores permitem discutir, em maior profundidade, o impacto de cada componente do framework proposto por meio de estudos de ablação. Em particular, foram analisadas quatro configurações principais:

- a) MASF-Swin treinado apenas com perda L1;
- b) MASF-Swin treinado com perda $\mathcal{L}_{perceptual}$;
- c) MASF-Mamba treinado com perda $\mathcal{L}_{perceptual}$;

- d) MASF-Swin refinado com perda $\mathcal{L}_{física}$, a partir do modelo com perda $\mathcal{L}_{perceptual}$;
- e) MASF-Swin Multitask (Combinação de $\mathcal{L}_{perceptual}$ e $\mathcal{L}_{física}$).

A comparação entre o Swin com L1 e o Swin com perda perceptual evidencia o papel da função de perda no comportamento do modelo. A variante com L1 apresenta ENL muito elevado, indicando supressão forte do *speckle*, mas com PSNR, SSIM e FOM mais baixos, além de valores de KS e KL elevados e maior variabilidade (Tabela 4 e Tabela 5). Essa combinação está associada à presença de artefatos e texturas artificiais, como ilustrado na Figura 22. Já o MASF-Swin com $\mathcal{L}_{perceptual}$ reduz o ENL para níveis intermediários (ainda superiores aos filtros clássicos), melhora PSNR e SSIM e apresenta entropia próxima à dos métodos estatísticos, resultando em texturas mais naturais e coerentes. A análise dos mapas de atenção e da correlação com a textura (Tabela 6) reforça essa interpretação: no modelo com $\mathcal{L}_{perceptual}$, os *gates* e o *backbone* Swin exibem correlações positivas moderadas com a variância local, indicando que a atenção se concentra justamente em regiões texturalmente ricas, em vez de realçar artefatos.

A variante Multitask não apresenta ganhos consistentes em relação ao modelo com perda perceptual simples. Em termos de métricas globais, mantém PSNR e SSIM relativamente baixos, ENL intermediário e valores de KS e KL ainda elevados, sem se destacar em nenhuma métrica específica (Tabela 4 e Tabela 5). Os coeficientes de correlação da Tabela 6 mostram relações fracas e, no *backbone*, até negativas entre atenção e textura, sugerindo que o acréscimo de tarefas auxiliares não levou a um alinhamento melhor entre atenção e estrutura da cena. Isso indica que, na concepção atual do *framework*, o treino multitarefa não trouxe benefícios em termos de qualidade da reconstrução.

A inclusão do *backbone* Mamba teve como motivação explorar uma alternativa baseada em modelos de espaço de estados, potencialmente mais eficiente na captura de dependências de longo alcance. Os resultados, porém, indicam que o MASF-Mamba com perda perceptual apresenta desempenho global semelhante ou ligeiramente inferior ao MASF-Swin em diversas métricas, com ENL elevado, mas PSNR/SSIM modestos e KS/KL altos. A análise qualitativa dos mapas de atenção (Figura 29) revela um padrão pouco estruturado no *bottleneck*, coerente com a natureza unidimensional do Mamba. A Tabela 6 mostra ainda correlações fracas ou negativas entre atenção e textura em *gates* iniciais e no *backbone*, sugerindo que o modelo não explora de forma eficiente a estrutura 2D da imagem, ao contrário do Swin. Dada a complexidade de implementação, o uso do Mamba não se justifica no cenário estudado.

Por fim, a comparação entre o Swin treinado com $\mathcal{L}_{perceptual}$ e o Swin refinado com $\mathcal{L}_{física}$ demonstra o impacto da segunda etapa de otimização não supervisionada, informada pela física do *speckle*. O refinamento promove melhorias claras em métricas de homogeneidade e aderência estatística (reduções em KS e KL), aproximando a distribuição das intensidades e a variância residual do comportamento esperado, ainda que sem superar os filtros clássicos nessa dimensão. Em contrapartida, o ENL diminui em relação ao modelo apenas supervisionado, refletindo a reintrodução controlada de variações de intensidade e, portanto, de textura espacial mais realista, como ilustrado na Figura 23. Do ponto de vista visual, observa-se que o refinamento preserva as principais feições estruturais, com leve aumento de ruído em áreas homogêneas.

De forma resumida, os estudos de ablação indicam que:

- a) A perda L1 isolada leva a soluções com ENL muito alto, mas PSNR/SSIM e FOM baixos, maior instabilidade estatística e surgimento de artefatos;
- b) A perda perceptual é essencial para obter um equilíbrio melhor entre supressão de ruído, preservação de estrutura e textura visualmente realista, conforme constatado pelas métricas, pelas comparações qualitativas e pela análise de atenção;
- c) O Swin oferece um melhor equilíbrio entre capacidade de representação, alinhamento espacial da atenção e custo computacional do que o Mamba e o Multitask para o conjunto de experimentos realizados;
- d) A etapa de refinamento contribui para consolidar os ganhos em coerência estatística e qualidade das texturas, com impacto limitado na fidelidade estrutural observada.

5 CONCLUSÃO

5.1 SÍNTESE DOS RESULTADOS

Este trabalho propôs e avaliou o framework MASF para supressão de *speckle* em imagens Sentinel-1 GRD, combinando alvos multitemporais físicos, uma arquitetura híbrida baseada em U-Net com backbones Swin Transformer/Mamba e um módulo de atenção SkipGate. Após a otimização de hiperparâmetros com Optuna para cada configuração avaliada, o modelo foi treinado e analisado com base em um conjunto de métricas quantitativas (fidelidade, homogeneidade, preservação de bordas, aderência estatística etc.) e em inspeções qualitativas em patches de validação do canal VV.

Na comparação com filtros clássicos (Lee, Frost, Quegan, Lee Sigma e Gamma MAP), o MASF apresentou valores de ENL superiores, indicando maior homogeneidade em áreas estáveis, ao mesmo tempo em que manteve a Entropia de Shannon em níveis próximos aos dos métodos estatísticos, indicando preservação de riqueza de textura. Em contrapartida, PSNR e SSIM foram, em média, inferiores aos dos filtros clássicos, refletindo uma relação entre redução agressiva de ruído e fidelidade radiométrica/estrutural. Dentro das variações de modelos, o MASF-Swin com perda perceptual ($\mathcal{L}_{perceptual}$) mostrou o melhor equilíbrio entre suavização e preservação de estrutura, produzindo texturas mais naturais que o modelo com L1 puro, enquanto o MASF-Swin refinado com perda física ($\mathcal{L}_{física}$) melhorou a coerência estatística (KS e KL) relativa às demais variantes, ainda que sem superar os filtros clássicos neste aspecto. Qualitativamente, as imagens geradas apresentaram fundo mais regularizado, poucos artefatos no modelo com perda perceptual e boa preservação das principais feições, com leve comprometimento de bordas muito finas.

Os estudos de ablação mostraram que a perda composta é importante para evitar soluções excessivamente borradas e com artefatos, que o *backbone* Swin se sobressai frente ao Mamba e a variante Multitask, e que a etapa de refinamento físico atua como um ajuste fino que reforça a aderência à física do *speckle*. A análise dos mapas de atenção e de sua correlação com medidas de textura indicou que os *gates* mais profundos passam a concentrar atenção em regiões estruturalmente relevantes, enquanto o Swin capta dependências de maior escala, fornecendo uma interpretação física consistente para o comportamento do modelo.

5.2 CONTRIBUIÇÕES DO TRABALHO

As principais contribuições deste trabalho podem ser sintetizadas da seguinte forma:

- a) Proposição de um framework multitemporal para redução de *speckle* em imagens Sentinel-1, integrando alvos físicos (média/combinações multitemporais) com uma arquitetura de aprendizagem profunda voltada à restauração de imagens SAR;
- b) Desenvolvimento e avaliação de uma arquitetura híbrida que combina U-Net, backbone Swin Transformer/Mamba e módulo SkipGate, articulada a uma função de perda composta, visando equilibrar supressão de ruído e preservação de estrutura;
- c) Definição de um procedimento experimental sistemático, com otimização de hiperparâmetros via Optuna, uso de múltiplas métricas quantitativas (fidelidade, homogeneidade, preservação de bordas, aderência estatística) e comparação controlada com filtros clássicos amplamente utilizados;
- d) Inclusão de uma etapa de refinamento informado por física, demonstrando na prática como restrições inspiradas na física do *speckle* podem complementar o treinamento supervisionado baseado em alvos multitemporais;
- e) Utilização de mapas de atenção (SkipGate, Swin e Mamba) como ferramenta de interpretação, evidenciando a relação entre atenção, textura e estrutura no contexto de imagens SAR.

5.3 LIMITAÇÕES E TRABALHOS FUTUROS

Apesar dos resultados promissores, o estudo apresenta limitações. Em primeiro lugar, a análise quantitativa baseou-se em um conjunto restrito de áreas de estudo e configurações de dados (canais VV/VH do Sentinel-1), condicionado por limitações de hardware. Além disso, a avaliação concentrou-se na etapa de treino/validação, sem testes intensos totalmente independentes, em diferentes sensores ou modos de aquisição. Do ponto de vista da arquitetura, apenas algumas variantes de *backbone* (Swin e Mamba) e de função de perda foram exploradas, e a etapa de refinamento físico foi implementada em uma configuração específica, sem variações mais amplas de modelos e restrições.

Como melhorias para trabalhos futuros, destacam-se:

- a) Ampliar o conjunto de áreas e biomas, incluindo novos contextos geomorfológicos e climáticos, bem como diferentes polarizações e outros sensores SAR;

b) Aprofundar a etapa de refinamento físico, testando diferentes formas de perda física, estratégias de treinamento não supervisionado e modelos que incorporem explicitamente parâmetros físicos do sistema SAR;

c) Investigar *backbones* adicionais e arquiteturas mais leves, buscando reduzir o custo computacional e tornar o MASF mais viável para aplicações operacionais comerciais;

d) Avaliar o impacto do framework em tarefas de alto nível, como classificação, detecção de mudanças ou mapeamento, de modo a quantificar o ganho de desempenho quando as imagens filtradas pelo MASF são usadas em fluxos de monitoramento.

Além disso, a variabilidade observada em métricas como KS e KL sugere explorar mecanismos explícitos de controle de consistência e incerteza, por exemplo por meio de mapas de incerteza, de funções de perda probabilísticas ou ainda de redes adversariais (GANs, capazes de restringir a geração de texturas a regiões estatisticamente plausíveis. Tais extensões poderiam reduzir a ocorrência de casos isolados com desempenho estatístico muito discrepante, tornando a supressão de ruído mais estável e confiável em diferentes cenas e condições.

REFERÊNCIAS

- AKIBA1, T. *et al.* Optuna: A Next-generation Hyperparameter Optimization Framework. In: ACM SIGKDD INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY & DATA MINING, 25., 2019. **Proceedings [...]**. 2019. p. 2623-2631. Disponível em: <https://arxiv.org/abs/1907.10902>
- ANGAL, A. *et al.* The use of the Sonoran Desert as a pseudo-invariant site for optical sensor cross-calibration and long-term stability monitoring. In: **2010 IEEE International Geoscience and Remote Sensing Symposium**, p. 1656-1659, 2010. Disponível em: <https://pubs.usgs.gov/publication/70157297>
- BARRON, J. **A General and Adaptive Robust Loss Function**. 2017. arXiv:1701.03077. Disponível em: <https://arxiv.org/pdf/1701.03077>
- BAUER-MARSCHALLINGER, B. *et al.* Toward Global Soil Moisture Monitoring with Sentinel-1: Harnessing Assets and Overcoming Obstacles. **IEEE Transactions on Geoscience and Remote Sensing**, v. 57, n. 7, p. 5200–5219, 2019. Disponível em: https://iris.cnr.it/retrieve/a77d3132-8074-4a47-b0c9-30147d9acc4a/prod_393396-doc_187910.pdf
- CUMMING, I. G.; WONG, F. H. Digital processing of synthetic aperture radar data: algorithms and implementation. Norwood, MA: Artech House, 2005. Disponível em: <https://us.artechhouse.com/Digital-Processing-of-Synthetic-Aperture-Radar-Data-P1549.aspx>
- DE BERG, Mark *et al.* **Computational geometry: algorithms and applications**. 3rd ed. Berlin: Springer, 2008. Disponível em: https://cimec.org.ar/foswiki/pub/Main/Cimec/GeometriaComputacional/DeBerg_-_Computational_Geometry_-_Algorithms_and_Applications_2e.pdf
- EL MORTAJI, N. *et al.* High resolution land cover mapping and crop classification in the Loukkos watershed (Northern Morocco): **An approach using SAR Sentinel-1 time series**. **Revista de Teledetección**, n. 60, p. 47-69, 2022. Disponível em: <https://polipapers.upv.es/index.php/raet/article/view/17426>
- FILIPPONI, F. Sentinel-1 GRD Preprocessing Workflow. In: MULTIDISCIPLINARY DIGITAL PUBLISHING INSTITUTE PROCEEDINGS, 2019. **Proceedings [...]**, v. 18, n. 1, p. 11, 2019. Disponível em: <https://www.mdpi.com/2504-3900/18/1/11>
- FROST, V. S. *et al.* A model for radar images and its application to adaptive digital filtering of multiplicative noise. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. PAMI-4, n. 2, p. 157–166, 1982. Disponível em: <https://ieeexplore.ieee.org/document/4767223>
- GDAL/OGR CONTRIBUTORS. **GDAL/OGR Geospatial Data Abstraction software Library**. Open Source Geospatial Foundation, 2023. Disponível em: <https://gdal.org/en/stable/>

GU, A.; DAO, T. **Mamba: Linear-Time Sequence Modeling with Selective State Spaces**. 2023. arXiv:2312.00752. Disponível em: <https://arxiv.org/pdf/2312.00752>

HARALICK, R. M.; SHANMUGAM, K.; DINSTEN, I. Textural features for image classification. **IEEE Transactions on Systems, Man, and Cybernetics**, v. SMC-3, n. 6, p. 610–621, 1973. Disponível em: https://haralick.org/book_chapters/TexturalFeatures.pdf

HUOT, E. *et al.* Image processing for multitemporal SAR images. In: EUROPEAN SYMPOSIUM ON REMOTE SENSING, SAR IMAGE ANALYSIS, MODELLING, AND TECHNIQUES, 1998, Barcelona. **Proceedings [...]**. Barcelona, Spain: EOS-SPIE, 1998. Disponível em: <https://doi.org/10.1117/12.331352>

ISOLA, P.; ZHU, J. Y.; ZHOU, T.; EFROS, A. A. Image-to-Image Translation with Conditional Adversarial Networks. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017**. Disponível em: https://openaccess.thecvf.com/content_cvpr_2017/papers/Isola_Image-To-Image_Translation_With_CVPR_2017_paper.pdf

JOHNSON, J.; ALAHI, A.; FEI-FEI, L. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In: **European Conference on Computer Vision (ECCV), 2016**. Disponível em: <https://arxiv.org/abs/1603.08155>

KELLNDORFER, J.; CARTUS, O.; BISHOP, J.; WALKER, W.; HOLECZ, F. Global seasonal Sentinel-1 interferometric coherence and backscatter data set. **Scientific Data**, v. 9, article number 73, 2022. Disponível em: <https://www.nature.com/articles/s41597-022-01189-6>

KULLBACK, S.; LEIBLER, R. A. On information and sufficiency. **The Annals of Mathematical Statistics**, v. 22, n. 1, p. 79–86, 1951. Disponível em: <https://projecteuclid.org/journals/annals-of-mathematical-statistics/volume-22/issue-1/On-Information-and-Sufficiency/10.1214/aoms/1177729694.full>

LEE, J. S. Digital image enhancement and noise filtering by use of local statistics. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. PAMI-2, n. 2, p. 165–168, 1980. Disponível em: <https://ieeexplore.ieee.org/document/4766994>

LEE, J. S. A simple speckle smoothing algorithm for synthetic aperture radar images. **IEEE Transactions on Systems, Man, and Cybernetics**, v. SMC-13, n. 1, p. 85–89, 1983. Disponível em: <https://ieeexplore.ieee.org/document/6313036>

LEE, J. S. Speckle Suppression and Analysis for Synthetic Aperture Radar Images. **Optical Engineering**, v. 25, n. 5, p. 636–643, 1986. Disponível em: <https://doi.org/10.1117/12.7973877>

LI, J.; PENG, C. Weighted residual network for SAR automatic target recognition with data augmentation. **Frontiers in Neurorobotics**, v. 17, 2023. Disponível em: <https://www.frontiersin.org/journals/neurorobotics/articles/10.3389/fnbot.2023.1298653>

LIU, Z. *et al.* Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In: IEEE/CVF INTERNATIONAL CONFERENCE ON COMPUTER VISION (ICCV), 2021. **Proceedings [...]**. p. 10012–10022, 2021. Disponível em: <https://arxiv.org/pdf/2103.14030>

LOPES, A.; TOUZI, R.; NEZRY, E. Adaptive speckle filters and scene heterogeneity. **IEEE Transactions on Geoscience and Remote Sensing**, v. 28, n. 6, p. 992–1000, 1990. Disponível em: <https://doi.org/10.1109/36.62623>

LOSHCHILOV, I.; HUTTER, F. Decoupled Weight Decay Regularization. In: INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS, 2019. **Proceedings [...]**. 2019. Disponível em: <https://arxiv.org/pdf/1711.05101>

MOREIRA, A. *et al.* A tutorial on synthetic aperture radar. **IEEE Geoscience and Remote Sensing Magazine**, v. 1, n. 1, p. 6–43, 2013. Disponível em: <https://core.ac.uk/download/pdf/31005519.pdf>

OLIVER, C.; QUEGAN, S. **Understanding synthetic aperture radar images**. Raleigh, NC: SciTech Publishing, 2004. Disponível em: <https://books.google.com/books?id=IeGKe40S77AC>

PRATT, W. K. **Digital Image Processing**. 4. ed. New York: John Wiley & Sons, 2007. Disponível em: <https://onlinelibrary.wiley.com/doi/book/10.1002/0470097434>

QUEGAN, S.; LE TOAN, T. Analysing multitemporal SAR images. In: SECOND LATINO-AMERICAN SEMINAR ON RADAR REMOTE SENSING: IMAGE PROCESSING TECHNIQUES, 1998, São Paulo. **Proceedings [...]**. São Paulo, Brazil: ESASP-434, 1998. p. 17–25. Disponível em: <http://marte.dpi.inpe.br/col/dpi.inpe.br/lise/2005/01.13.18.00/doc/pcshaun.pdf>

QUEGAN, S.; YU, J. J. Filtering of multichannel SAR images. **IEEE Transactions on Geoscience and Remote Sensing**, v. 39, n. 11, p. 2375–2391, 2001. Disponível em: https://www.researchgate.net/publication/3202751_Filtering_of_multichannel_SAR_images

RAISSI, M.; PERDIKARIS, P.; KARNIADAKIS, G. E. Physics-informed neural networks: A Deep Learning framework for solving forward and inverse problems involving nonlinear partial differential equations. **Journal of Computational Physics**, v. 378, p. 686–707, 2019. Disponível em: <https://arxiv.org/pdf/1711.10561>

RONNEBERGER, O.; FISCHER, P.; BROX, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In: MEDICAL IMAGE COMPUTING AND COMPUTER-ASSISTED INTERVENTION – MICCAI 2015. **Proceedings [...]**. Springer, 2015. p. 234–241. Disponível em: <https://arxiv.org/pdf/1505.04597>

SHANNON, C. E. A mathematical theory of communication. **The Bell System Technical Journal**, v. 27, n. 3, p. 379–423, 1948. Disponível em: <https://people.math.harvard.edu/~ctm/home/text/others/shannon/entropy/entropy.pdf>

SMARA, Y.; MANSOURA, I. S. A study of multitemporal filtering techniques for SAR images speckle reduction. In: GOOSSENS, R. (ed.). **Remote Sensing in Transition**.

Rotterdam: Millpress, 2004. p. 135–141. Disponível em:
<https://www.earsel.org/symposia/2003-symposium-Ghent/pdf/B14.pdf>

ULLMANN, T. *et al.* Assessing Spatiotemporal Variations of Sentinel-1 InSAR Coherence at Different Time Scales over the Atacama Desert, Chile (2015–2018). **Remote Sensing**, v. 11, n. 24, p. 2960, 2019. Disponível em: <https://www.mdpi.com/2072-4292/11/24/2960>

WANG, Z.; BOVIK, A. C.; SHEIKH, H. R.; SIMONCELLI, E. P. Image quality assessment: from error visibility to structural similarity. **IEEE Transactions on Image Processing**, v. 13, n. 4, p. 600–612, 2004. Disponível em: <https://ece.uwaterloo.ca/~z70wang/research/ssim/>

ZHANG, K. *et al.* Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising. **IEEE Transactions on Image Processing**, v. 26, n. 7, p. 3142–3155, 2017. Disponível em: <https://arxiv.org/abs/1608.03981>

GLOSSÁRIO

Abertura Sintética: Técnica utilizada em sistemas SAR para simular uma antena fisicamente muito longa, utilizando o movimento da plataforma de aquisição para alcançar alta resolução espacial.

AdamW: Um algoritmo de otimização para treinamento de redes neurais que melhora a regularização do otimizador Adam ao desacoplar o decaimento de peso da atualização do gradiente.

Amostragem por Rejeição: Uma técnica de Monte Carlo para gerar observações de uma distribuição. No contexto do *framework*, significa gerar pontos aleatórios em uma área maior e descartar ("rejeitar") aqueles que não cumprem um critério específico (neste caso, estar dentro do polígono da zona segura).

Aumento de Dados ou Augmentação de Dados (*Data Augmentation*): Técnica utilizada para aumentar artificialmente o tamanho de um conjunto de dados de treinamento, criando cópias modificadas dos dados existentes.

Backbone: Em arquiteturas de redes neurais, refere-se à rede principal de extração de características (encoder e bottleneck no contexto da U-Net)

Baseline (Método de Referência): Um algoritmo ou modelo padrão, bem estabelecido, contra o qual o desempenho de novas propostas é comparado para demonstrar melhoria.

Batch (Lote): Em *machine learning*, um subconjunto do conjunto de dados de treinamento que é processado em uma única iteração do algoritmo de otimização.

Bottleneck (Gargalo): A camada mais profunda de uma arquitetura codificador-decodificador, onde a representação dos dados possui a menor resolução espacial e o maior nível de abstração semântica.

Campo de Deslocamento: Uma matriz ou grade que armazena os vetores de deslocamento (Δx , Δy) para cada pixel ou bloco de uma imagem, utilizados para corrigir desalinhamentos geométricos.

Campo Receptivo: A região na imagem de entrada que influencia a ativação de uma única unidade em uma camada mais profunda da rede.

Casco Convexo (*Convex Hull*): O menor polígono convexo que contém um conjunto de pontos no plano. É uma estrutura fundamental em geometria computacional para determinar o contorno de uma nuvem de pontos.

Codificador (*Encoder*): O caminho de contração em uma arquitetura de rede neural como a U-Net, responsável por extrair mapas de características e reduzir a resolução espacial.

Coefficiente de Variação: Medida de dispersão estatística, definida como a razão entre o desvio padrão e a média. Em imagens SAR, é usado para quantificar a heterogeneidade local.

Convolução Transposta: Operação em redes convolucionais que aumenta a resolução espacial de um mapa de características, frequentemente utilizada no decodificador de arquiteturas como a U-Net.

Corregistro: Processo de alinhar geometricamente duas ou mais imagens da mesma área para que os pixels correspondam à mesma localização no terreno.

Dados Inválidos (*No-data*): Pixels em uma imagem de satélite que não contêm informação válida sobre o terreno, geralmente representados por valores de zero ou nulos.

Decodificador (*Decoder*): O caminho de expansão em uma arquitetura de rede neural como a U-Net, responsável por aumentar a resolução espacial e reconstruir a saída a partir dos mapas de características.

Deep Learning (Aprendizado Profundo): Subcampo do aprendizado de máquina baseado em redes neurais artificiais com múltiplas camadas (profundas), capaz de aprender representações complexas de dados.

Distância de Kolmogorov-Smirnov: Uma medida da distância entre duas funções de distribuição acumulada.

Divergência de Kullback-Leibler: Uma medida da diferença entre duas distribuições de probabilidade.

Entropia (de Shannon): Uma medida da incerteza ou aleatoriedade em um conjunto de dados. Em processamento de imagens, é frequentemente usada para quantificar a complexidade ou a quantidade de "informação" em uma imagem.

Estabilidade Temporal: No contexto de sensoriamento remoto para a geração de alvos, refere-se a áreas onde a estrutura física do terreno e a cobertura do solo permanecem relativamente constantes ao longo do tempo, permitindo que as variações no sinal sejam atribuídas principalmente ao ruído ou a efeitos sazonais, e não a mudanças antrópicas.

Estimação a Posteriori Máxima (MAP): Método estatístico Bayesiano para estimar uma quantidade desconhecida, que busca maximizar a probabilidade da estimativa dados os valores observados.

Estudo de Ablação: Em aprendizado de máquina, um processo experimental que consiste em remover sistematicamente componentes de um modelo ou algoritmo para avaliar a contribuição de cada um para o desempenho geral.

Fator de Amortecimento (*Damping Factor*): Parâmetro utilizado em filtros (como o de Frost) que controla o grau de suavização, ponderando a influência dos pixels vizinhos.

Filtro Multitemporal: Técnica de processamento que utiliza uma série de imagens da mesma área, adquiridas em diferentes momentos, para reduzir o ruído, baseando-se na redundância temporal do sinal.

Framework: Estrutura de suporte conceitual e tecnológico que serve como base para a construção de aplicações, fornecendo um conjunto de ferramentas e bibliotecas.

Função de Perda (*Loss Function*): Função matemática que quantifica a diferença (erro) entre a previsão de um modelo e o valor real. O objetivo do treinamento é minimizar esta função.

Gating (Mecanismo de Portão): Em redes neurais, uma técnica que utiliza uma unidade multiplicativa (geralmente com ativação Sigmoid) para controlar o fluxo de informação através de uma conexão.

GDAL: Uma biblioteca de software de código aberto para a leitura e escrita de formatos de dados geoespaciais, amplamente utilizada em aplicações de sensoriamento remoto e geoprocessamento.

Generalização: Em *machine learning*, a capacidade de um modelo de performar bem em dados novos e não vistos, que não fizeram parte do seu conjunto de treinamento.

GeoTIFF: Um padrão de metadados de domínio público que permite que informações de georreferenciamento sejam embutidas em um arquivo de imagem TIFF.

Gradiente Evanesciente: Problema em redes neurais profundas onde os gradientes se tornam progressivamente menores à medida que retropropagam pelas camadas, dificultando o treinamento das camadas iniciais.

Ground Range Detected (GRD): Nível de processamento de dados SAR do Sentinel-1, em que a imagem foi projetada da geometria de aquisição para a geometria de solo.

Ground Truth: Em *machine learning*, refere-se aos dados de referência ou "a verdade" que o modelo tenta prever. No contexto deste trabalho, é a imagem-alvo "limpa".

Hiperparâmetro: Um parâmetro de configuração cujo valor é definido antes do início do processo de treinamento e que não é aprendido pela rede (*e.g.*, taxa de aprendizado, tamanho do lote, pesos da função de perda).

Imagens Escravas: Imagens que serão alinhadas à imagem de referência (imagem mestre) no processo de corregristo.

Imagem Mestre: Em um processo de corregristo, a imagem de referência à qual todas as outras imagens são alinhadas.

Invariância: Em *machine learning*, a propriedade de um modelo de produzir a mesma saída mesmo quando a entrada é submetida a certas transformações (*e.g.*, um modelo de reconhecimento de objetos deve ser invariante à rotação).

Kernel de Ponderação: Uma matriz de pesos utilizada em operações de filtragem, onde cada peso determina a influência de um pixel vizinho no cálculo do valor do pixel central.

Mamba: Arquitetura de rede neural baseada em SSMs que processa dados sequenciais com complexidade linear, apresentando-se como uma alternativa eficiente aos *Transformers*.

Mapa de Características (*Feature Map*): A saída de uma camada convolucional, representando a presença de certas características (como bordas ou texturas) na imagem de entrada.

Matriz de Coocorrência de Níveis de Cinza (GLCM): Uma matriz que descreve a frequência com que pares de pixels com valores de intensidade específicos ocorre em uma imagem a uma determinada distância e orientação, servindo como base para métricas de textura.

Normalização (Z-score): Processo de reescalar os dados para que tenham uma média de 0 e um desvio padrão de 1. Essencial para a estabilidade do treinamento de redes neurais.

On-the-fly: Em computação, refere-se a operações que são realizadas conforme necessário, no momento da execução, em vez de serem pré-calculadas e armazenadas.

Optuna: Uma biblioteca de software para otimização automática de hiperparâmetros, projetada para ser eficiente e flexível.

Otimizador: Algoritmo (*e.g.*, Adam, AdamW) que ajusta os pesos de uma rede neural com base nos gradientes calculados pela função de perda, a fim de minimizar o erro.

Overfitting (Sobreajuste): Fenômeno que ocorre quando um modelo de aprendizado de máquina se ajusta excessivamente aos dados de treinamento, perdendo a capacidade de generalizar para novos dados não vistos.

Perda Composta: Uma função de perda que é a soma ponderada de múltiplos componentes de perda, permitindo a otimização simultânea de diferentes objetivos.

Polarização: A orientação do campo elétrico de uma onda eletromagnética. Em SAR, refere-se à polarização das ondas transmitidas e recebidas (e.g., VV, VH).

Pooling: Operação que reduz a resolução espacial dos mapas de características, resumindo as informações de uma vizinhança (e.g., *max-pooling* seleciona o valor máximo).

PyTorch: Uma biblioteca de *machine learning* de código aberto baseada em *Torch*, usada para aplicações como visão computacional e processamento de linguagem natural.

Pytest: Um *framework* de código aberto que facilita a criação e automação de testes para aplicações *python*.

Python: Uma linguagem interpretada multiparadigma de alto nível amplamente utilizada no contexto de *machine learning* devido à sua sintaxe amigável, bibliotecas extensas e forte suporte comunitário.

Qualidade Radiométrica: Medida da fidelidade com que os valores de pixel em uma imagem representam o retroespalhamento real do alvo no solo, livre de distorções e ruídos.

Regularização: Conjunto de técnicas (ex: *weight decay*, *dropout*) utilizadas para prevenir o *overfitting*, adicionando uma penalidade à complexidade do modelo.

Regularização Indutiva: Uma forma de regularização onde a arquitetura do modelo ou a tarefa de treinamento é projetada de forma a enviesar o modelo para aprender soluções que se alinham com o conhecimento prévio sobre o problema.

ReLU (*Rectified Linear Unit*): Uma função de ativação não-linear que introduz dispersão nas representações da rede, definida como $f(x) = \max(0, x)$. Ela retorna o próprio valor de entrada se ele for positivo, e zero caso contrário.

Retroespalhamento (*Backscatter*): A porção do sinal de radar que é refletida por um alvo e retorna na direção do sensor.

Ruído Residual: A diferença entre a imagem de entrada (ruidosa) e a saída do modelo (filtrada). Em um cenário ideal, este resíduo deve conter apenas o ruído e nenhum sinal da cena original.

Ruído Speckle ou Speckle: Ruído granular multiplicativo, inerente às imagens SAR, resultante da interferência construtiva e destrutiva dos sinais refletidos por múltiplos meios físicos dentro de uma célula de resolução.

Sigma Naught ou Sigma Zero (σ^0): Coeficiente de retroespalhamento normalizado, que representa a refletividade do alvo por unidade de área no solo. É uma medida utilizada para comparar o brilho de alvos em diferentes imagens SAR.

Sigmoid: Uma função de ativação não-linear que mapeia qualquer valor real para o intervalo $[0, 1]$. É frequentemente utilizada em mecanismos de *gating* ou em camadas de saída para predição de probabilidades.

Skip Connection: Conexão em uma rede neural que "pula" uma ou mais camadas, ligando diretamente a saída de uma camada anterior a uma camada posterior. Na U-Net, conecta o *encoder* ao *decoder*.

SkipGate: Módulo customizado implementado no projeto que aplica um mecanismo de atenção à *skip connection*, permitindo que a rede aprenda a ponderar a importância das informações que fluem através dela.

Swin Transformer: Uma arquitetura de *Vision Transformer* hierárquica que calcula a autoatenção dentro de janelas locais e deslocadas, permitindo uma modelagem eficiente de dependências espaciais com complexidade computacional linear.

Tradução de Imagem para Imagem: Classe de problemas em visão computacional onde o objetivo é transformar uma imagem de um domínio para outro (*e.g.*, de uma imagem com ruído para uma sem ruído).

Treinamento Não Supervisionado: Um paradigma de treinamento de *machine learning* onde o modelo aprende a partir de dados de entrada não rotulados, identificando padrões ou estruturas inerentes aos dados sem uma saída correta predefinida.

Treinamento Supervisionado: O paradigma de treinamento mais comum, onde o modelo aprende a partir de um conjunto de dados que contém pares de entrada e saída correta (rótulo ou *ground truth*), aprendendo a mapear um no outro.

U-Net: Arquitetura de rede neural convolucional com uma estrutura simétrica de codificador-decodificador e conexões de atalho (*skip connections*), amplamente utilizada para tarefas de segmentação e restauração de imagens.

APÊNDICE A – PRODUTOS DO SENTINEL-1 UTILIZADOS

Localização	Produtos utilizados
Deserto de Atacama (Chile)	S1A_IW_GRDH_1SDV_20250409T100201_20250409T100226_058678_074416_D68F S1A_IW_GRDH_1SDV_20250421T100201_20250421T100226_058853_074B42_4BD5 S1A_IW_GRDH_1SDV_20250503T100201_20250503T100226_059028_07524C_8689 S1A_IW_GRDH_1SDV_20250515T100201_20250515T100226_059203_0758AA_46F4 S1A_IW_GRDH_1SDV_20250527T100200_20250527T100225_059378_075EC3_E79A S1A_IW_GRDH_1SDV_20250608T100200_20250608T100225_059553_0764C0_676F S1A_IW_GRDH_1SDV_20250620T100159_20250620T100224_059728_076AC7_E8AE S1A_IW_GRDH_1SDV_20250702T100158_20250702T100223_059903_0770E2_3F61 S1A_IW_GRDH_1SDV_20250714T100158_20250714T100223_060078_0776E5_49F0
Deserto de Negev (Israel)	S1A_IW_GRDH_1SDV_20250331T033631_20250331T033656_058543_073E95_9DBB S1A_IW_GRDH_1SDV_20250412T033631_20250412T033656_058718_0745BF_9DDA S1A_IW_GRDH_1SDV_20250424T033631_20250424T033656_058893_074CE9_26EA S1A_IW_GRDH_1SDV_20250506T033631_20250506T033656_059068_0753E1_55C8 S1A_IW_GRDH_1SDV_20250518T033631_20250518T033656_059243_075A04_5D50 S1A_IW_GRDH_1SDV_20250530T033630_20250530T033655_059418_07601C_BC3E S1A_IW_GRDH_1SDV_20250611T033630_20250611T033655_059593_076610_FDAF S1A_IW_GRDH_1SDV_20250623T033629_20250623T033654_059768_076C1A_B4CB S1A_IW_GRDH_1SDV_20250705T033628_20250705T033653_059943_077235_04CA
Deserto de Sonora (EUA)	S1A_IW_GRDH_1SDV_20250324T013325_20250324T013359_058440_073A64_4B53 S1A_IW_GRDH_1SDV_20250405T013325_20250405T013359_058615_07417C_D632 S1A_IW_GRDH_1SDV_20250417T013325_20250417T013359_058790_0748A2_0EA2 S1A_IW_GRDH_1SDV_20250511T013325_20250511T013359_059140_075673_762A S1A_IW_GRDH_1SDV_20250523T013325_20250523T013358_059315_075C84_D348 S1A_IW_GRDH_1SDV_20250604T013324_20250604T013358_059490_076294_C684 S1A_IW_GRDH_1SDV_20250616T013323_20250616T013357_059665_07688C_FC12 S1A_IW_GRDH_1SDV_20250628T013323_20250628T013357_059840_076E9E_45EF S1A_IW_GRDH_1SDV_20250710T013322_20250710T013356_060015_0774AE_494E
Outback Australiano (Austrália)	S1B_IW_GRDH_1SDV_20210909T204302_20210909T204327_028626_036A9B_9A7B S1B_IW_GRDH_1SDV_20210921T204302_20210921T204327_028801_036FF8_1080 S1B_IW_GRDH_1SDV_20211003T204303_20211003T204328_028976_037540_94AF S1B_IW_GRDH_1SDV_20211015T204303_20211015T204328_029151_037A9D_C0F7 S1B_IW_GRDH_1SDV_20211027T204303_20211027T204328_029326_03800E_DA62 S1B_IW_GRDH_1SDV_20211108T204302_20211108T204327_029501_038560_A895 S1B_IW_GRDH_1SDV_20211120T204302_20211120T204327_029676_038AC0_OEE6 S1B_IW_GRDH_1SDV_20211202T204302_20211202T204327_029851_039048_3EF2 S1B_IW_GRDH_1SDV_20211214T204301_20211214T204326_030026_0395D1_D888
Tóquio (Japão)	S1A_IW_GRDH_1SDV_20251127T204344_20251127T204409_062068_07C411_ABD2 S1A_IW_GRDH_1SDV_20251115T204345_20251115T204410_061893_07BD39_3BF5 S1A_IW_GRDH_1SDV_20251103T204346_20251103T204411_061718_07B671_1EA1 S1A_IW_GRDH_1SDV_20251022T204346_20251022T204411_061543_07AF83_97B1 S1A_IW_GRDH_1SDV_20251010T204346_20251010T204411_061368_07A878_B622 S1A_IW_GRDH_1SDV_20250928T204346_20250928T204411_061193_07A16B_2FC3 S1A_IW_GRDH_1SDV_20250916T204346_20250916T204411_061018_079A5E_E21D S1A_IW_GRDH_1SDV_20250904T204345_20250904T204410_060843_07935B_02E7 S1A_IW_GRDH_1SDV_20250823T204345_20250823T204410_060668_078C6B_6F3A

Mar de Aral (Uzbequistão)	S1A_IW_GRDH_1SDV_20251206T021015_20251206T021040_062188_07C8BF_96AA S1A_IW_GRDH_1SDV_20251124T021016_20251124T021041_062013_07C1EA_4399 S1A_IW_GRDH_1SDV_20251112T021017_20251112T021042_061838_07BB17_259D S1A_IW_GRDH_1SDV_20251031T021018_20251031T021043_061663_07B454_991D S1A_IW_GRDH_1SDV_20251019T021018_20251019T021043_061488_07AD46_F127 S1A_IW_GRDH_1SDV_20251007T021018_20251007T021043_061313_07A641_1936 S1A_IW_GRDH_1SDV_20250925T021018_20250925T021043_061138_079F34_2736 S1A_IW_GRDH_1SDV_20250913T021018_20250913T021043_060963_07982C_6A79 S1A_IW_GRDH_1SDV_20250901T021017_20250901T021042_060788_07912C_0BDE
Patagônia, Chubut (Argentina)	S1C_IW_GRDH_1SDV_20251208T094149_20251208T094218_005358_00AA7E_29BD S1C_IW_GRDH_1SDV_20251126T094149_20251126T094219_005183_00A48E_17E7 S1C_IW_GRDH_1SDV_20251114T094150_20251114T094219_005008_009EA4_062C S1C_IW_GRDH_1SDV_20251102T094150_20251102T094220_004833_0098D6_DF13 S1C_IW_GRDH_1SDV_20251021T094151_20251021T094220_004658_00935D_9790 S1C_IW_GRDH_1SDV_20251009T094151_20251009T094220_004483_008E0D_7333 S1C_IW_GRDH_1SDV_20250927T094150_20250927T094220_004308_0088BB_7E07 S1C_IW_GRDH_1SDV_20250915T094150_20250915T094219_004133_00836E_90A6 S1C_IW_GRDH_1SDV_20250903T094150_20250903T094219_003958_007E19_F3C5
Koue Bokkeveld (África do Sul)	S1B_IW_GRDH_1SDV_20190215T033640_20190215T033716_014951_01BEAA_AE90 S1B_IW_GRDH_1SDV_20190203T033640_20190203T033716_014776_01B8F3_2A45 S1B_IW_GRDH_1SDV_20190122T033641_20190122T033716_014601_01B34B_C212 S1B_IW_GRDH_1SDV_20181229T033641_20181229T033717_014251_01A801_0A77 S1B_IW_GRDH_1SDV_20181205T033642_20181205T033718_013901_019C79_23BC S1B_IW_GRDH_1SDV_20181123T033643_20181123T033718_013726_0196CF_6216 S1B_IW_GRDH_1SDV_20181111T033643_20181111T033719_013551_019148_9F3E S1B_IW_GRDH_1SDV_20181030T033643_20181030T033719_013376_018BD3_8BC0 S1B_IW_GRDH_1SDV_20181006T033643_20181006T033719_013026_0180FB_A9F3
Rocky Mountain, Colorado (EUA)	S1A_IW_GRDH_1SDV_20251128T130946_20251128T131011_062078_07C47E_13BB S1A_IW_GRDH_1SDV_20251116T130947_20251116T131012_061903_07BDA6_1782 S1A_IW_GRDH_1SDV_20251104T130948_20251104T131013_061728_07B6DE_2C29 S1A_IW_GRDH_1SDV_20251023T130948_20251023T131013_061553_07AFEF_5FF4 S1A_IW_GRDH_1SDV_20251011T130949_20251011T131014_061378_07AE6E_A6B5 S1A_IW_GRDH_1SDV_20250929T130948_20250929T131013_061203_07A1D9_B6FE S1A_IW_GRDH_1SDV_20250917T130948_20250917T131013_061028_079ACC_ECE7 S1A_IW_GRDH_1SDV_20250905T130948_20250905T131013_060853_0793C6_6ED1 S1A_IW_GRDH_1SDV_20250824T130948_20250824T131013_060678_078CD8_83DB
Estreito de Gibraltar (Marrocos)	S1A_IW_GRDH_1SDV_20230225T181800_20230225T181825_047396_05B085_19C9 S1A_IW_GRDH_1SDV_20230309T181800_20230309T181825_047571_05B66E_042D S1A_IW_GRDH_1SDV_20230321T181800_20230321T181825_047746_05BC5E_80E9 S1A_IW_GRDH_1SDV_20230402T181800_20230402T181825_047921_05C236_4735 S1A_IW_GRDH_1SDV_20230414T181801_20230414T181826_048096_05C825_5B62 S1A_IW_GRDH_1SDV_20230426T181801_20230426T181826_048271_05CE05_CD76 S1A_IW_GRDH_1SDV_20230508T181802_20230508T181827_048446_05D3DF_659A S1A_IW_GRDH_1SDV_20230520T181803_20230520T181828_048621_05D909_6DEC S1A_IW_GRDH_1SDV_20230601T181803_20230601T181828_048796_05DE3F_DCDC

APÊNDICE B – PARÂMETROS DO MODELO

Modelo	Parâmetro e valor adotado
MASF-Swin (L1)	patch_size: 512 patches_per_image: 200 model_type: swin in_chans: 6 out_chans: 2 embed_dim: 96 swin_depth: 3 swin_heads: 4 window_size: 8 use_skip_gates: True perceptual_augmentation: True physical_augmentation: True perceptual_loss: - type: l1 weight: 1.0 - type: l2 weight: 0.0 - type: charbonnier weight: 0.0 - type: ssim weight: 0.0 - type: gradient weight: 0.0 - type: frequency weight: 0.0 physical_loss: - type: speckle_variance weight: 0.5 params: - kernel_size: 9 - type: speckle_mean weight: 0.25 - type: histogram weight: 0.25 params:

	bins: 256 min_val: -5.0 max_val: 5.0 train: optimizer: "adam" use_multitask_train: false batch_size: 2 num_epochs: 150 learning_rate: 0.00029168438660425404 weight_decay: 0.000014130544093928408 num_workers: 4 seed: 14
MASF-Swin ($\mathcal{L}_{perceptual}$)	patch_size: 512 patches_per_image: 200 model_type: swin in_chans: 6 out_chans: 2 embed_dim: 96 swin_depth: 3 swin_heads: 4 window_size: 8 use_skip_gates: True perceptual_augmentation: True physical_augmentation: True perceptual_loss: - type: l1 weight: 0.0 - type: l2 weight: 0.0 - type: charbonnier weight: 0.5 - type: ssim weight: 0.2 - type: gradient weight: 0.15 - type: frequency weight: 0.15 physical_loss:

	- type: speckle_variance - weight: 0.5 - params: - kernel_size: 9 - type: speckle_mean - weight: 0.25 - type: histogram - weight: 0.25 - params: - bins: 256 - min_val: -5.0 - max_val: 5.0 - train: - optimizer: "adam" - use_multitask_train: false - batch_size: 2 - num_epochs: 150 - learning_rate: 0.00037805283863190283 - weight_decay: 0.002790527067529427 - num_workers: 4 - seed: 14
MASF-Mamba ($\mathcal{L}_{perceptual}$)	- patch_size: 512 - patches_per_image: 200 - model_type: swin - in_chans: 6 - out_chans: 2 - embed_dim: 96 - mamba_depth: 3 - mamba_d_state: 16 - mamba_d_conv: 4 - mamba_expand: 4 - use_skip_gates: True - perceptual_augmentation: True - physical_augmentation: True - perceptual_loss: - type: l1 - weight: 0.0 - type: l2

	weight: 0.0 - type: charbonnier weight: 0.5 - type: ssim weight: 0.2 - type: gradient weight: 0.15 - type: frequency weight: 0.15 physical_loss: - type: speckle_variance weight: 0.5 params: kernel_size: 9 - type: speckle_mean weight: 0.25 - type: histogram weight: 0.25 params: bins: 256 min_val: -5.0 max_val: 5.0 train: optimizer: "adam" use_multitask_train: false batch_size: 2 num_epochs: 150 learning_rate: 0.0009684762914968829 weight_decay: 0.001676563343956272 num_workers: 4 seed: 14
MASF-Swin Multitask	patch_size: 512 patches_per_image: 200 model_type: swin in_chans: 6 out_chans: 2 embed_dim: 96 swin_depth: 3

	<pre> swin_heads: 4 window_size: 8 use_skip_gates: True perceptual_augmentation: True physical_augmentation: True perceptual_loss: - type: l1 weight: 0.0 - type: l2 weight: 0.0 - type: charbonnier weight: 0.5 - type: ssim weight: 0.2 - type: gradient weight: 0.15 - type: frequency weight: 0.15 physical_loss: - type: speckle_variance weight: 0.5 params: kernel_size: 9 - type: speckle_mean weight: 0.25 - type: histogram weight: 0.25 params: bins: 256 min_val: -5.0 max_val: 5.0 train: optimizer: "adam" use_multitask_train: false batch_size: 2 num_epochs: 150 learning_rate: 0.00037805283863190283 weight_decay: 0.002790527067529427 num_workers: 4 </pre>
--	--

	seed: 14 multitask: reconstruction_weight: 0.9 noise_mask_weight: 0.1 enable_reconstruction: true enable_noise_mask: true head_depth: 2
MASF-Swin Refinado	patch_size: 512 patches_per_image: 200 model_type: swin in_chans: 6 out_chans: 2 embed_dim: 96 swin_depth: 3 swin_heads: 4 window_size: 8 use_skip_gates: True perceptual_augmentation: True physical_augmentation: True perceptual_loss: - type: l1 weight: 0.0 - type: l2 weight: 0.0 - type: charbonnier weight: 0.5 - type: ssim weight: 0.2 - type: gradient weight: 0.15 - type: frequency weight: 0.15 physical_loss: - type: speckle_variance weight: 0.5 params: kernel_size: 9

	- type: speckle_mean weight: 0.25 - type: histogram weight: 0.25 params: - bins: 256 - min_val: -5.0 - max_val: 5.0 train: - optimizer: "adam" - use_multitask_train: false - batch_size: 2 - num_epochs: 150 - learning_rate: 0.00037805283863190283 - weight_decay: 0.002790527067529427 - num_workers: 4 - seed: 14 refinement: - use_multitask_refinement: true - learning_rate: 5.0e-07 - weight_decay: 0
--	---

APÊNDICE C – REPOSITÓRIO DO CÓDIGO-FONTE DO MASF

O código-fonte desenvolvido neste trabalho encontra-se disponibilizado em repositório público no GitHub, incluindo instruções de instalação, arquivos de configuração e exemplos de uso do *framework* MASF.

Endereço: <https://github.com/RonaldoGorgulho/MASF>