



UNIVERSIDADE  
ESTADUAL DE LONDRINA

---

RAFAEL LUCIEN BAHR ARIAS

SISTEMA ADAPTATIVO DE DETECÇÃO E REMOÇÃO  
DE RUÍDOS EM VÍDEOS

RAFAEL LUCIEN BAHR ARIAS

SISTEMA ADAPTATIVO DE DETECÇÃO E REMOÇÃO  
DE RUÍDOS EM VÍDEOS

Dissertação apresentada ao Programa de Mestrado em Ciência da Computação da Universidade Estadual de Londrina para obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Alan Salvany Felinto

Londrina  
2019

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UEL

Arias, Rafael.

Sistema adaptativo de detecção e remoção de ruídos em vídeos / Rafael Arias. - Londrina, 2019.  
71 f. : il.

Orientador: Alan Felinto.

Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Londrina, Centro de Ciências Exatas, Programa de Pós-Graduação em Ciência da Computação, 2019.

Inclui bibliografia.

1. Processamento de vídeos - Tese. 2. Detecção de ruído - Tese. 3. Estimacão de ruído - Tese. 4. Remocão de ruído - Tese. I. Felinto, Alan. II. Universidade Estadual de Londrina. Centro de Ciências Exatas. Programa de Pós-Graduação em Ciência da Computação. III. Título.

RAFAEL LUCIEN BAHR ARIAS

**SISTEMA ADAPTATIVO DE DETECÇÃO E REMOÇÃO DE  
RUÍDOS EM VÍDEOS**

Dissertação apresentada ao Programa de Mestrado em Ciência da Computação da Universidade Estadual de Londrina para obtenção do título de Mestre em Ciência da Computação.

**BANCA EXAMINADORA**

---

Orientador: Prof. Dr. Alan Salvany Felinto  
Universidade Estadual de Londrina – UEL

---

Prof. Dr. Osvaldo Coelho Pereira Neto  
Universidade Estadual de Londrina – UEL

---

Prof. Dr. Sylvio Barbon Junior  
Universidade Estadual de Londrina – UEL

Londrina, 21 de março de 2019.

*Este trabalho é dedicado à minha família,  
que sempre me deu suporte durante a  
trajetória até aqui.*

## AGRADECIMENTOS

Gostaria de agradecer à minha família, sobretudo meus pais, e a todos que em algum momento contribuíram para a minha formação.

Em especial gostaria de agradecer ao professor Dr. Alan Salvany Felinto, que me ajudou imensamente e sempre me motivou a solucionar os problemas que surgiam.

Agradeço à CAPES pela concessão da bolsa durante o período de realização deste mestrado.

*“A perfeição não é alcançada quando não  
há mais nada a ser incluído, mas quando  
não há mais nada a ser retirado.”  
(Antoine de Saint-Exupéry)*

ARIAS, R. L. B. **Sistema adaptativo de detecção e remoção de ruídos em vídeos.** 2019. 71 f. Dissertação (Mestrado em Ciência da Computação) – Universidade Estadual de Londrina, Londrina, 2019.

## RESUMO

Os vídeos digitais proporcionaram avanços fundamentais na era da informação, mas ainda enfrentam problemas decorrentes das falhas originadas no seu processo de aquisição e transmissão, que ocasionam ruídos e a perda das informações presentes no vídeo. Os algoritmos de restauração geralmente presumem um único modelo de ruído, como por exemplo o aditivo Gaussiano, enquanto na prática as imagens e vídeos podem ser corrompidos por diferentes tipos de ruído que seguem uma distribuição distinta dependendo da aplicação, como o ruído impulsivo e o multiplicativo speckle. Neste trabalho, a proposta é de uma abordagem adaptativa que leva em consideração estas três classes de ruído para a restauração de vídeos, realizando a detecção e estimação do nível de ruído com modelos de aprendizado supervisionado a fim de adaptar a técnica de filtragem, sendo empregada uma estratégia de pré-processamento com fusão de atributos para aprimorar a identificação do ruído incluindo a sua ausência. Foram avaliadas seis técnicas de redução de ruído em doze bases de dados de vídeos utilizando a métrica de qualidade VQM (Video Quality Metric), uma métrica específica para vídeos que leva em consideração a dimensão temporal e apresenta alta correlação com a avaliação humana. A partir disso, o sistema de detecção e estimação foi avaliado em relação à seleção da técnica de filtragem e ao desempenho da restauração resultante em comparação a outros três estimadores de ruído da literatura. Os resultados obtidos mostraram que a filtragem selecionada pelo sistema proposto foi quantitativamente superior na maioria dos tipos de ruídos considerados, onde a remoção de ruído obteve resultados relativamente próximos da seleção de referência e substancialmente melhores do que na filtragem sem adaptação.

**Palavras-chave:** Remoção de ruído. Vídeo. Detecção. Estimação

ARIAS, R. L. B. **Adaptive system for video noise detection and removal**. 2019. 71 p. Dissertation (Master's in Science in Computer Science) – Universidade Estadual de Londrina, Londrina, 2019.

## ABSTRACT

Digital videos have provided fundamental breakthroughs in the information age, but they still face problems originated from flaws in their acquisition and transmission process, which cause noise and loss of information in the video. Denoising algorithms often assume a single noise model, for instance, additive Gaussian noise, whereas in practice images and videos can be corrupted by different types of noise that follow a distinct distribution depending on the application, such as impulsive and multiplicative speckle noise. In this work, the proposal is an adaptive approach that takes into account these three noise classes for the restoration of videos, by performing noise detection and estimation with supervised learning models in order to adapt the filtering technique, employing a preprocessing strategy with feature fusion to improve the identification of noise including its absence. Six denoising techniques were evaluated in twelve video databases using the Video Quality Metric (VQM), a video-specific metric that takes into account the temporal dimension and has a high correlation with human evaluation. Based on the experiments, the detection and estimation system was evaluated in relation to the selection of the filtering technique and the performance of the resulting restoration in comparison to three other noise estimators from the literature. The obtained results showed that the selected filtering by the proposed system was quantitatively superior in the majority of the considered noise types, where the noise removal obtained results relatively close to the reference selection and substantially better than in the filtering without adaptation.

**Keywords:** Denoising. Video. Detection. Estimation.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Fluxograma do sistema adaptativo de detecção e remoção de ruídos. . . . .	32
Figura 2 – Bases de dados VQEG ANSI, <i>Influence Content</i> , IVP e LIVE. . . . .	34
Figura 3 – Bases de dados <i>Dorr</i> , <i>MMSys</i> , PEV, <i>Video Trace</i> e <i>VQEG Phase I</i> . . . . .	34
Figura 4 – Base selecionada pelo YouTube-8M contendo 32 categorias de vídeos. . . . .	35
Figura 5 – Vídeos corrompidos com diferentes níveis de ruído impulsivo. . . . .	36
Figura 6 – Vídeos corrompidos com diferentes níveis de ruído Gaussiano. . . . .	36
Figura 7 – Vídeos corrompidos com diferentes níveis de ruído <i>speckle</i> . . . . .	36
Figura 8 – Fluxograma para avaliação dos algoritmos de redução de ruído. . . . .	37
Figura 9 – Abordagem de pré-processamento do vídeo com filtragem e fusão de atributos. . . . .	39
Figura 10 – Comparação dos algoritmos de filtragem para redução de ruído Gaussiano. . . . .	42
Figura 11 – Comparação visual da redução de ruído Gaussiano com $\sigma = 40$ . . . . .	43
Figura 12 – Comparação dos algoritmos de filtragem para redução de ruído impulsivo. . . . .	44
Figura 13 – Comparação visual da redução de ruído impulsivo com $\sigma = 40$ . . . . .	44
Figura 14 – Comparação dos algoritmos de filtragem para redução de ruído <i>speckle</i> . . . . .	45
Figura 15 – Comparação visual da redução de ruído <i>speckle</i> com $\sigma = 40$ . . . . .	46
Figura 16 – Variação do F1 para cada classe em relação ao nível de ruído. . . . .	50
Figura 17 – Estimação do nível de ruído com coeficiente de correlação de Pearson de 95%. . . . .	52
Figura 18 – Vídeo com ruído <i>speckle</i> ( $\sigma = 20$ ) processado pelo VBM3D com diferentes níveis como parâmetro de entrada. . . . .	55
Figura 19 – Vídeo com ruído Gaussiano ( $\sigma = 50$ ) processado pelo VBM3D com diferentes níveis como parâmetro de entrada. . . . .	56
Figura 20 – Comparação da redução de ruído pela métrica VQM com seleção da técnica de filtragem utilizando detecção (SVM) e estimação (GBoost) de ruído. . . . .	62

## LISTA DE TABELAS

Tabela 1 – Técnicas de redução de ruído e respectivos parâmetros considerados nos experimentos. . . . .	37
Tabela 2 – Atributos selecionados na primeira fase e as respectivas importâncias. . . . .	39
Tabela 3 – Desvio padrão das pontuações VQM obtidas para cada ruído e técnica de filtragem. . . . .	47
Tabela 4 – Comparação dos algoritmos de classificação para aprendizado supervisionado com <i>10-fold Cross-Validation</i> na detecção de ruído. . . . .	48
Tabela 5 – Matriz de confusão normalizada das classes de detecção de ruído com acurácia geral de 96.45% (modelo SVM). . . . .	48
Tabela 6 – Comparação da detecção de ruído para cada classe. . . . .	49
Tabela 7 – Comparação da detecção de ruído para cada nível (SVM). . . . .	49
Tabela 8 – Comparação dos algoritmos de regressão para aprendizado supervisionado com <i>10-fold Cross-Validation</i> na estimação de ruído. . . . .	51
Tabela 9 – Comparação do modelo de regressão do GBoost em relação a PCA, DCT e WT na estimação do nível de ruído para diferentes classes. . . . .	53
Tabela 10 – Comparação da redução de ruído com VBM3D utilizando os níveis estimados como parâmetro de filtragem. . . . .	54
Tabela 11 – Comparação do número de amostras para cada técnica de filtragem de referência de acordo com o tipo de ruído. . . . .	57
Tabela 12 – Comparação da seleção da técnica de filtragem com diferentes métodos de detecção e estimação de ruído. . . . .	58
Tabela 13 – Matriz de confusão da seleção da filtragem pela detecção de ruído (SVM) e o nível real do ruído. . . . .	59
Tabela 14 – Matriz de confusão da seleção da filtragem pelo tipo real do ruído e o nível estimado de ruído (GBoost). . . . .	59
Tabela 15 – Matriz de confusão da seleção da filtragem pela detecção de ruído (SVM) e o nível estimado de ruído (GBoost). . . . .	59
Tabela 16 – Comparação do parâmetro com o valor de referência na seleção correta da técnica de filtragem de ruído. . . . .	60
Tabela 17 – Comparação da redução de ruído pela métrica VQM com seleção da técnica de filtragem usando diferentes métodos de detecção e estimação de ruído. . . . .	61

## LISTA DE ABREVIATURAS E SIGLAS

BFM	<i>Brightness Flicking Metric</i>
DCT	<i>Discrete Cosine Transform</i>
GBoost	<i>Gradient Boosting</i>
HSV	<i>Hue, Saturation, Value</i>
K-NN	<i>K-Nearest Neighbors</i>
MAD	<i>Median Absolute Deviation</i>
MLP	<i>Multilayer Perceptron</i>
MOS	<i>Mean Opinion Score</i>
MSE	<i>Mean Squared Error</i>
NLM	<i>Non-Local Means</i>
PCA	<i>Principal Component Analysis</i>
PCC	<i>Pearson Correlation Coefficient</i>
PSNR	<i>Peak Signal-To-Noise Ratio</i>
RF	<i>Random Forest</i>
ROAD	<i>Rank-Ordered Absolute Differences</i>
SSIM	<i>Structural Similarity</i>
SVM	<i>Support Vector Machine</i>
TPR	<i>True Positive Rate</i>
VBM3D	<i>Video Block-Matching and 3-Dimensional Filtering</i>
VQM	<i>Video Quality Metric</i>
WT	<i>Wavelet Transform</i>

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>14</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>16</b>
<b>2.1</b>	<b>Processamento Digital de Imagens e Vídeos</b>	<b>16</b>
2.1.1	Modelos de Cores	16
2.1.2	Histograma	17
<b>2.2</b>	<b>Ruídos</b>	<b>18</b>
2.2.1	Ruído Gaussiano	18
2.2.2	Ruído Impulsivo	18
2.2.3	Ruído <i>Speckle</i>	19
<b>2.3</b>	<b>Algoritmos de Filtragem</b>	<b>19</b>
2.3.1	Filtro Gaussiano	20
2.3.2	Filtro de Mediana	21
2.3.3	Filtro Bilateral	21
2.3.4	Filtro de Difusão Anisotrópica	22
2.3.5	<i>Non-Local Means</i>	22
2.3.6	<i>Video Block-Matching 3D</i>	23
<b>2.4</b>	<b>Métricas de Qualidade</b>	<b>24</b>
2.4.1	Métricas de Referência Completa	24
2.4.1.1	<i>Mean Squared Error</i>	24
2.4.1.2	<i>Peak Signal-To-Noise Ratio</i>	25
2.4.1.3	<i>Structural Similarity</i>	25
2.4.1.4	<i>Video Quality Metric</i>	26
2.4.2	Métricas Sem Referência	26
2.4.2.1	<i>Rank-Ordered Absolute Differences</i>	27
2.4.2.2	<i>Brightness Flicking Metric</i>	27
2.4.2.3	<i>Blurriness</i>	27
<b>3</b>	<b>TRABALHOS RELACIONADOS</b>	<b>29</b>
<b>4</b>	<b>PROCEDIMENTOS METODOLÓGICOS</b>	<b>32</b>
<b>4.1</b>	<b>Aquisição</b>	<b>32</b>
<b>4.2</b>	<b>Avaliação das técnicas de redução de ruído</b>	<b>35</b>
<b>4.3</b>	<b>Extração e seleção de atributos</b>	<b>38</b>
<b>4.4</b>	<b>Pré-processamento e fusão de atributos</b>	<b>38</b>
<b>4.5</b>	<b>Aprendizado supervisionado</b>	<b>40</b>

5	RESULTADOS E DISCUSSÃO . . . . .	41
5.1	Avaliação das técnicas de filtragem de ruído . . . . .	41
5.2	Detecção do tipo de ruído . . . . .	47
5.3	Estimação do nível de ruído . . . . .	51
5.4	Redução de ruído com seleção da técnica de filtragem . . . . .	57
6	CONCLUSÕES E TRABALHOS FUTUROS . . . . .	64
	REFERÊNCIAS . . . . .	65
	Trabalhos Publicados pelo Autor . . . . .	71

# 1 INTRODUÇÃO

Nos últimos anos, a tecnologia avançou para um ponto onde a aquisição, transmissão e reprodução de vídeos digitais se tornaram bastante práticas e eficientes. Existe uma gama de aplicações que se beneficiam desta evolução, como as videoconferências, compartilhamento de vídeos na internet, televisão digital, procedimentos médicos, etc. [1]. Porém, ainda existem situações nas quais as imagens e vídeos gravados sofrem degradações significantes, como em condições precárias de ambiente, falhas nos sensores de gravação e dispositivos de transmissão, interferências de sinal, até problemas causados por algoritmos de compressão [2]. Assim, a restauração de imagens e vídeos sempre foi um campo essencial da computação gráfica, não apenas para aprimorar a qualidade visual como também para aumentar o desempenho de etapas posteriores de processamento, como análise e reconhecimento do conteúdo.

A atenuação ou redução de ruídos é uma etapa crítica nas aplicações de processamento gráfico, já que busca isolar as informações relevantes das interferências externas [3]. Os ruídos muitas vezes são originados em condições naturais como a baixa iluminação, que juntamente da alta sensibilidade de uma câmera de baixo custo resulta em uma gravação com grande propensão a ruídos [4]. Vídeos tendem a apresentar ainda mais ruídos que imagens devido à elevada velocidade de captura dos quadros ou *frames* pela câmera [5], mas por outro lado, a vantagem é que vídeos possuem redundância temporal, ou seja, a informação presente em cada quadro de vídeo se repete por vários quadros adjacentes. Dessa forma, os algoritmos de redução de ruído em vídeos podem tirar proveito disso e restaurar falhas de uma maneira mais eficaz [6]. Métodos de filtragem de imagens também são aplicáveis, mas esta abordagem denominada *intraframe* considera cada frame como temporalmente independente, e pode obter resultados insuficientes além de gerar novos artefatos temporais [1].

Apesar de diversas técnicas terem sido propostas para a remoção de ruído, a natureza do problema depende do tipo de ruído que interferiu no sinal. As técnicas de restauração geralmente presumem um único modelo, como por exemplo o ruído aditivo Gaussiano, que segue uma distribuição normal de valores de degradação, o que frequentemente não ocorre na prática [5, 7]. Outros ruídos comuns são originados de fontes como sistemas de quantização, laser e radar, que podem assumir modelos de distribuição distintos como o ruído impulsivo, também conhecido como “sal e pimenta”, que assume valores extremos nos pixels [8]. Além de ruídos aditivos, cujo componente de degradação é independente do sinal, há também o ruído multiplicativo que é correlacionado com o conteúdo, como o ruído *speckle* [9]. Dessa maneira, é fundamental detectar e identificar corretamente os tipos de ruído que estão presentes, permitindo a restauração com técnicas

mais apropriadas.

Este trabalho propõe um sistema adaptativo para remoção do ruído aditivo Gaussiano, impulsivo e o ruído multiplicativo *speckle* em vídeos. São avaliadas seis técnicas de redução de ruído, sendo estas o filtro Gaussiano, filtro de mediana, bilateral, anisotrópico, filtro de médias não-locais (NLM) e *Video Block-Matching 3D* (VBM3D) em doze bases de dados de vídeos utilizando a métrica de qualidade VQM (*Video Quality Metric*) para comparar o vídeo processado com a versão original sem degradações e determinar a melhor filtragem para cada instância de ruído.

Para detectar o tipo do ruído presente ou a sua ausência e estimar o nível ou intensidade do ruído representada pelo desvio padrão ( $\sigma$ ), são utilizados modelos de aprendizado supervisionado para classificação e regressão respectivamente, comparando seis algoritmos: *K-Nearest Neighbors* (KNN), árvore de decisão C4.5, *Support Vector Machine* (SVM), rede neural *Multilayer Perceptron* (MLP), *Random Forest* (RF) e *Gradient Boosting* (GBoost). O desempenho da classificação é comparado por meio de métricas como acurácia, precisão, taxa de verdadeiros positivos (TPR) e pontuação F1, enquanto a regressão é avaliada pela média e desvio padrão das diferenças absolutas, o erro quadrático médio (MSE) e o coeficiente de correlação de Pearson (PCC).

Os atributos selecionados dos modelos de predição incluem uma série de medidas estatísticas ou métricas de processamento de imagem e vídeo: média, desvio padrão, desvio absoluto mediano (MAD), entropia, assimetria e curtose do histograma, *Rank-Ordered Absolute Differences* (ROAD), métrica *blurriness*, *blockiness* e de oscilação do brilho (BFM). Para aprimoramento dos modelos, é empregada uma estratégia de pré-processamento com filtragem e fusão dos atributos.

A estimação do nível de ruído é comparada a outros estimadores baseados em análise de componentes principais (PCA), transformada discreta de cosseno (DCT) e transformada *wavelet* (WT). A partir da detecção e estimação de ruído, é avaliada a seleção da técnica e do parâmetro associado de filtragem em relação às técnicas de referência definidas para cada tipo e nível reais do ruído, e o desempenho da redução de ruído resultante com a seleção do sistema em relação à seleção de referência, pelos demais estimadores e à filtragem sem seleção.

O trabalho está organizado da seguinte forma: o capítulo 2 apresenta a fundamentação teórica necessária para a compreensão do trabalho; no capítulo 3 são discutidos os trabalhos relacionados, comparando outras abordagens para a detecção, estimação e avaliação da filtragem de diferentes tipos de ruído; o capítulo 4 descreve os procedimentos metodológicos utilizados para a realização dos experimentos, incluindo métodos empregados no desenvolvimento do sistema; os resultados obtidos são apresentados e discutidos no capítulo 5 e por fim as considerações finais no capítulo 6.

## 2 FUNDAMENTAÇÃO TEÓRICA

Para a realização deste trabalho, foi necessário o estudo de múltiplos tópicos relacionados à área de processamento digital da Computação Gráfica, que são apresentados nas diferentes seções deste capítulo.

### 2.1 Processamento Digital de Imagens e Vídeos

Na área de processamento de imagens, uma imagem pode ser definida como uma função bidimensional  $f(x, y)$ , onde  $x$  e  $y$  são coordenadas do plano e o valor de  $f$  para qualquer par de coordenadas é denominado intensidade ou nível de cinza da imagem no ponto  $(x, y)$  [10]. Quando as coordenadas e intensidades são discretas e finitas, a imagem é chamada de imagem digital. Imagens e vídeos digitais nada mais são do que imagens e vídeos que foram convertidos para um formato binário passível de ser lido por um computador. Assim, o campo de processamento digital refere-se aos processos realizados por um computador digital em imagens ou vídeos digitais [1].

Um vídeo analógico é representado como um sinal contínuo variante no tempo, expresso por  $I(x, y, t)$  onde  $(x, y)$  são coordenadas espaciais contínuas e  $t$  é o tempo contínuo. Um vídeo digital é uma sequência de imagens digitais, denominadas *frames* ou quadros, sendo obtido por amostragem e quantização do vídeo analógico, ou por amostragem direta nos sensores de gravação. A amostragem é um processo de conversão do sinal contínuo do espaço-tempo para um sinal discreto, enquanto a quantização refere-se à conversão de um espaço de valores contínuos para valores discretos em um intervalo finito. É possível observar que durante a amostragem para vídeo digital existe perda de informação, já que as mudanças ao longo do tempo não podem ser completamente representadas em um conjunto discreto [1].

Uma imagem digital é formada por finitos elementos denominados *pixels*, onde cada um possui uma localização e valores específicos [10]. Na quantização são calculados os valores de intensidade dos pixels dentro do intervalo discreto e finito, resultando em uma imagem em níveis de cinza se ela é representada apenas pelos valores de intensidade. Para imagens coloridas, há mais de um valor associado a cada pixel, para isso a câmera ou dispositivo de captura utiliza sensores distintos [1]. Nesse caso, a cor do pixel é definida por um modelo ou espaço de cor.

#### 2.1.1 Modelos de Cores

Um modelo de cor é uma tupla de valores que descreve uma cor específica, podendo ser representada como um ponto no espaço. O propósito na utilização de diferentes espaços

de cores é facilitar a sua especificação de acordo com a necessidade, seja em hardware, como câmeras, impressoras e monitores, ou aplicações em computação gráfica, em um formato padronizado e aceito por todos [11].

Um modelo bastante comum em televisões, computadores e câmeras digitais é o RGB (*Red, Green, Blue*), onde cada cor é formada pela adição das cores primárias vermelho, verde e azul [10]. O problema deste modelo para os algoritmos de processamento é que ele não separa as componentes de luminância e crominância. A luminância pode ser percebida como a intensidade ou brilho dos pixels, estando muito mais próxima dos níveis de cinza da imagem, enquanto a enquanto a crominância refere-se à informação das cores dos pixels, denominada matiz ou tonalidade [1]. Existem vários modelos de cores que permitem trabalhar com a luminância e crominância separadamente, como o HSV (*Hue, Saturation, Value*), que separa a cor nos componentes de matiz, saturação e brilho, sendo bastante utilizada em pesquisas devido à sua naturalidade [12].

A maioria dos algoritmos para imagens e vídeos digitais são aplicados em imagens de valor ou canal único. Por isso, em imagens coloridas as técnicas são aplicadas separadamente em um ou mais canais, considerando cada componente como uma imagem separada e em seguida recombinao os resultados. Pode-se verificar que o modelo RGB contém muitas informações sobrepostas, enquanto em modelos como o HSV é possível isolar os dados de intensidade no canal de luminância, assim o processamento de muitos algoritmos é realizado após a conversão para um destes espaços e voltando para o espaço original em RGB [1].

### 2.1.2 Histograma

O histograma é uma função discreta  $h(r_k) = n_k$  onde  $r_k$  é o  $k$ -ésimo valor de intensidade no intervalo  $[0, L - 1]$  e  $n_k$  é o número de pixels da imagem com intensidade  $r_k$  [10]. Em outras palavras, é uma relação entre cada valor possível do pixel e sua ocorrência na imagem. A análise da distribuição do histograma indica a condição de visibilidade de uma cena: caso a contagem de pixels no histograma esteja concentrada em uma determinada região, significa que as imagens estão com pouco contraste ou brilho. Uma maior variação na distribuição da contagem no histograma indica imagens com bom ajuste de contraste, permitindo melhor distinção entre os objetos e detalhes [13].

A partir do histograma, pode-se calcular a função  $p(r_k) = \frac{n_k}{n}$  onde  $n$  é o número total de pixels na imagem. Esta função fornece a probabilidade de um pixel da imagem apresentar a intensidade  $r_k$ . Assim, pode-se observar que um histograma representa uma função de distribuição de probabilidades, apresentando informações importantes da imagem além de seguir certas propriedades, como por exemplo a soma dos valores de  $p(r_k)$  que deve ser necessariamente igual a 1 [11].

## 2.2 Ruídos

O ruído pode ser definido como um sinal que interfere na transmissão ou medida de outro sinal. Ele está presente em quase todos os ambientes, e dependendo da sua origem pode ser classificado em diversas categorias, como ruído acústico, térmico, eletrostático ou eletromagnético [7]. Em imagens e vídeos os ruídos podem surgir durante a gravação, processamento e transmissão dos dados [4]. Uma imagem ou quadro de vídeo deteriorado por ruído pode ser representado pela equação (2.1) [14]:

$$v(i) = u(i) + n(i), \quad (2.1)$$

onde  $v(i)$  é o valor observado,  $u(i)$  é o valor “verdadeiro” e  $n(i)$  é o valor do ruído no pixel  $i$ .

O valor do ruído pode ser puramente aleatório, quando todas as frequências do sinal possuem a mesma probabilidade de ocorrer com certa intensidade. Neste caso, ele é denominado “ruído branco” [7]. Porém, na maioria dos casos, os ruídos que ocorrem naturalmente podem ser modelados ou aproximados por uma função densidade de probabilidade, que descreve a probabilidade relativa da ocorrência de cada valor no intervalo do sinal, de forma semelhante ao histograma descrito na seção 2.1.2. No caso do ruído branco, a distribuição é constante com autocorrelação zero.

Um ruído definido por uma função de distribuição aleatória pode ainda variar em relação à intensidade indicada pela variância ou desvio padrão ( $\sigma$ ), enquanto a média dos valores está relacionada à sua distribuição [9]. Alguns dos principais tipos de ruído são apresentados a seguir.

### 2.2.1 Ruído Gaussiano

O ruído aditivo Gaussiano é um dos tipos mais considerados que se aproxima da distribuição de probabilidade normal, sendo causado por fontes naturais como baixa iluminação e alta temperatura, assim, é utilizado para modelar ruídos térmicos, e em certos casos também ruídos de fótons e granulação de filme [1]. Também é chamado de ruído eletrônico, pois surge devido a problemas eletrônicos em amplificadores ou detectores [8]. Nas imagens e vídeos geralmente é modelado com média tendendo a zero, ou seja, distribuição centrada em zero com oscilação de valores positivos e negativos [4].

### 2.2.2 Ruído Impulsivo

O ruído impulsivo é também conhecido como ruído “sal e pimenta” pois os pixels corrompidos assumem valor mínimo ou máximo, tornando-se pretos ou brancos, respectivamente [1]. Pode ocorrer durante a transmissão devido à interferência eletromagnética e problemas na conversão de dados analógicos para digitais. A imagem não fica totalmente corrompida, outros pixels ficam inalterados [7].

### 2.2.3 Ruído *Speckle*

O ruído *speckle* é um ruído multiplicativo que ocorre em sistemas de laser, radar ou acústicos, estando relacionado aos ruídos “salpicados” que seguem a distribuição gama de probabilidade [9]. Ao contrário do ruído aditivo, a deterioração é dependente e correlacionada ao conteúdo, como representado na equação (2.2):

$$v(i) = u(i) + u(i) * n(i), \quad (2.2)$$

onde  $v(i)$  é o valor observado,  $u(i)$  é o valor “verdadeiro” e  $n(i)$  é o valor do ruído no pixel  $i$ .

## 2.3 Algoritmos de Filtragem

Os métodos para filtragem de imagens e vídeos podem ser classificados em duas categorias principais: os que atuam no domínio do espaço, e no domínio da frequência ou transformação [3]. No caso de vídeos, surge um novo domínio, denominado temporal ou espaço-temporal [15].

Na redução de ruído, como um vídeo é uma sequência de imagens ou quadros, os métodos para imagem também são aplicáveis, porém este tipo de abordagem chamado de filtragem *intraframe* considera que cada quadro é temporalmente independente. Ignorando a correlação existente na dimensão temporal, os resultados obtidos tem grande chance de não serem ótimos, podendo também gerar novos artefatos (temporais) no vídeo [1]. Assim, os métodos mais eficazes consistem na exploração do domínio temporal, visando aproveitar-se de informações redundantes presentes em quadros adjacentes do vídeo para a correção de sua qualidade. Estes métodos são denominados *interframe* [15].

As técnicas de filtragem espacial operam diretamente na matriz de pixels da imagem, possuindo uma forma geral dada pela equação (2.3) [10]:

$$g(x, y) = T[f(x, y)], \quad (2.3)$$

onde  $g(x, y)$  é a imagem processada,  $f(x, y)$  é a imagem original, e  $T$  é um operador em  $f$  definido sobre uma vizinhança de cada pixel  $(x, y)$ .

Dependendo do funcionamento, os filtros espaciais podem ainda ser classificados em lineares ou não-lineares, e locais ou não-locais. Os filtros lineares utilizam algoritmos com funções lineares ou operadores de espaço, com efeito uniforme por todos os pixels da imagem, não preservando bordas ou detalhes. Já os filtros não-lineares estão relacionados a operadores de ponto, que são mais seletivos e geram diferentes efeitos em regiões distintas da imagem [1]. A localidade se refere ao grupo de pixels considerados no processamento de cada pixel; se apenas a região na vizinhança do pixel é levada em conta, o filtro é

considerado local, mas se pixels de outros segmentos da imagem podem ser utilizados, o filtro é denominado não-local [3].

Já a filtragem no domínio da frequência é baseada no teorema da convolução. Se  $g(x, y)$  é a imagem formada pela convolução da imagem  $f(x, y)$  com um operador linear  $h(x, y)$ , onde  $*$  é o operador de convolução na equação (2.4) [10]:

$$g(x, y) = f(x, y) * h(x, y), \quad (2.4)$$

pelo teorema da convolução, a relação no domínio da frequência também é válida na equação (2.5):

$$G(u, v) = F(u, v)H(u, v), \quad (2.5)$$

onde  $G$ ,  $F$  e  $H$  são as transformadas de Fourier de  $g$ ,  $f$  e  $h$ , respectivamente. Dessa forma, a filtragem espacial pode ser identificada pela utilização de máscaras espaciais no processamento, enquanto a filtragem no domínio da frequência opera sobre a transformada de Fourier da imagem [11].

Em relação ao efeito, é possível classificar os filtros em passa-baixa, passa-alta e passa-faixa. Bordas, detalhes e ruídos da imagem estão relacionados aos componentes de alta frequência, por isso os filtros denominados “passa-baixa” possuem um efeito de borramento ao atenuar ou eliminar as componentes de alta frequência no domínio da transformada de Fourier. Já a filtragem passa-alta atenua os elementos de baixa frequência, assim realçando as bordas e outras regiões de alto contraste da imagem. Os filtros do tipo passa-faixa podem suavizar componentes acima da sua frequência de corte superior e abaixo do corte inferior, sendo utilizados para procedimentos específicos na área de processamento [10].

### 2.3.1 Filtro Gaussiano

Um dos filtros passa-baixa mais simples é o filtro de média, que pode ser definido por uma máscara com todos os coeficientes iguais, geralmente de valor igual a um. Uma máscara de tamanho 3 por 3 pixels fará com que o novo valor de cada pixel seja a média do conjunto do seu valor com o de seus 8 vizinhos, ou menos no caso das extremidades da imagem [11].

Já no filtro gaussiano, a diferença é que este utiliza uma máscara gaussiana, onde os valores próximos do centro da máscara são maiores e os valores nas extremidades são menores, de acordo com a distribuição normal gaussiana. A resposta de um filtro gaussiano para um pixel na posição  $(m, n)$  pode ser definida pela equação (2.6), onde  $\sigma$  é o desvio padrão da distribuição [1]:

$$g(m, n) = \frac{1}{2\pi\sigma^2} e^{-\frac{m^2+n^2}{2\sigma^2}} \quad (2.6)$$

Como é possível observar, tanto o filtro de média quanto o gaussiano são filtros lineares e locais, cujo problema na remoção de ruído é a incapacidade de preservar bordas e detalhes finos da imagem [3].

### 2.3.2 Filtro de Mediana

No caso do filtro de mediana, uma técnica simples que também é um filtro local, porém não-linear, o operador de mediana que é aplicado nos valores da vizinhança do pixel alvo. Dado o conjunto ordenado  $x(n)$  para  $i = 1, 2, \dots, N$  valores na janela de observação, a saída  $y(n)$  é dada pela equação (2.7) [1]:

$$y(n) = \begin{cases} x(\frac{n+1}{2}), & \text{se } n \text{ é ímpar} \\ \frac{x(\frac{n}{2}) + x(\frac{n}{2}+1)}{2}, & \text{se } n \text{ é par} \end{cases} \quad (2.7)$$

O filtro de mediana contorna o problema dos filtros lineares permitindo a preservação de bordas e detalhes para máscaras pequenas, onde pelo menos metade da janela contém pixels de borda.

### 2.3.3 Filtro Bilateral

O filtro bilateral também é uma técnica local e não-linear eficaz na preservação de bordas, que consiste em calcular o pixel alvo como uma média ponderada dos pixels vizinhos, onde o valor do peso é maior para os pixels mais próximos tanto em posição quanto na intensidade do valor [16]. A média ponderada é realizada em uma vizinhança  $S$  ao redor do centro  $s_0 = \{x_0, y_0\}$  com intensidade  $I(s_0)$ , e pode ser expressa pela equação (2.8) [17]:

$$O(s_0) = \frac{\sum_{s \in S} f(s, s_0) I(s)}{\sum_{s \in S} f(s, s_0)}, \quad (2.8)$$

onde a função  $f$  é a filtragem para cada vizinho  $s$  ao redor de  $s_0$ , definida pela equação (2.9):

$$f(s, s_0) = g_s(s - s_0) \cdot g_t(I(s) - I(s_0)), \quad (2.9)$$

com  $g_s$  e  $g_t$  representando filtros gaussianos para os pesos da posição e intensidade dos pixels, respectivamente (2.10):

$$g_s(s) = g(x, \sigma_s)g(y, \sigma_s) \quad \text{e} \quad g_t(I) = g(I, \sigma_t), \quad (2.10)$$

e  $\sigma_s$  e  $\sigma_t$  são parâmetros para a posição e intensidade dos vizinhos. Quando o valor de  $\sigma_t$  é maior, pixels com intensidades mais distantes do alvo são considerados na média ponderada, contanto que estejam próximos na posição pelo  $\sigma_s$ . Se o valor de  $\sigma_s$  é maior, a intensidade de pixels cuja posição está mais distante do alvo serão considerados na média ponderada, de acordo com a proximidade da intensidade pelo  $\sigma_t$  [16].

### 2.3.4 Filtro de Difusão Anisotrópica

A filtragem de difusão anisotrópica, técnica chamada também de Perona-Malik, o nome de seus criadores, possui o objetivo de reduzir ruído sem afetar partes importantes da imagem como as bordas e os detalhes [16]. Foi desenvolvida a partir de uma interpretação do filtro Gaussiano como uma equação diferencial parcial da condução de calor, com  $\frac{\partial I}{\partial t} = -\nabla \cdot [g(\|\nabla I\|) \nabla I]$  onde  $I$  é a intensidade de cada pixel, sendo considerada como o calor da equação que é propagado para os vizinhos de acordo com a variação espacial do calor. A partir disso foi introduzida uma função de difusão para as bordas que varia a condutância de acordo com o gradiente da imagem, prevenindo a “passagem de calor” pelas bordas, como expresso na equação (2.11) [18]:

$$\frac{\partial I}{\partial t} = \text{div}[g(\|\nabla I\|) \nabla I], \quad (2.11)$$

onde  $\text{div}$  é o operador de divergência. Para a função de difusão das bordas  $g(x)$  foram propostas duas expressões:

$$g_1(x) = \frac{1}{1 + \frac{x^2}{\sigma^2}} \quad \text{e} \quad g_2(x) = e^{-\frac{x^2}{\sigma^2}}, \quad (2.12)$$

onde  $\sigma$  é o parâmetro de escala em relação à intensidade, que especifica em qual gradiente a difusão deve ser interrompida, permitindo controlar a quantidade de borramento em relação aos aspectos da imagem que não fazem parte das bordas mais nítidas. Apesar da eficiência na preservação das bordas, a complexidade do processo de difusão do filtro anisotrópico torna a técnica mais lenta [18].

### 2.3.5 *Non-Local Means*

O filtro de médias não-locais ou *Non-Local Means* (NLM) proposto por Buades et al. foi uma inovação em relação à maioria dos métodos de redução de ruído como o filtro bilateral e anisotrópico, que operam localmente na imagem [14]. A ideia do NLM é restaurar um pixel realizando uma estimativa de todos os outros pixels da imagem, ponderados de forma proporcional à semelhança com os vizinhos do pixel analisado. Em outras palavras, é uma média ponderada de todos os pixels onde o peso está relacionado à similaridade com os vizinhos do pixel alvo, como é possível visualizar na fórmula da equação (2.13) [19]:

$$NL(v)(i) = \sum_{j \in I} w(i, j) v(j), \quad (2.13)$$

onde  $NL(v)(i)$  é o valor de restauração do pixel  $i$ ,  $j \in I$  representa a iteração em todos os valores de intensidade da imagem, e  $w(i, j)$  é o peso que representa a quantidade de semelhança entre os vizinhos do pixel  $i$  e de cada um dos outros pixels. A função de similaridade utilizada é geralmente a distância euclidiana, de forma que os pesos são

computados pela fórmula da equação (2.14):

$$w(i, j) = \frac{1}{Z(i)} e^{-\frac{\|v(N_i) - v(N_j)_{2,a}\|^2}{h^2}}, \quad (2.14)$$

onde  $Z(i)$  é um fator de normalização que equivale a  $\sum_j w(i, j)$ , e  $h$  é um parâmetro para controlar o valor dos pesos. Quando  $h$  é maior, a diferença entre os pesos é reduzida e mais pixels são considerados para restaurar o pixel alvo, tornando a imagem mais borrada e com menos detalhes. Se  $h$  for pequeno, os pesos tem valor relevante apenas para os pixels bastante similares ao alvo, atenuando menos ruído mas preservando os detalhes.

Em uma imagem com  $M$  pixels, são computados  $M^2$  pesos, o que torna o algoritmo clássico impraticável, por isso foram realizadas otimizações para reduzir consideravelmente o número de pesos calculados e diminuir a sua complexidade, com a redução do espaço de busca e cálculo prévio da média de vários grupos de pixels, descartando os que possuem valor muito diferente do alvo [19]. O algoritmo NLM também foi intuitivamente adaptado para vídeos como um método *interframe* ao incluir os pixels dos quadros adjacentes no cálculo do pixel alvo de um dado quadro, produzindo melhores resultados do que em uma execução *intraframe* [20].

Apesar da complexidade, o NLM é mais eficiente que vários métodos para vídeo, uma vez que não precisa realizar estimação de movimento devido à natureza não-local do algoritmo, pois como foi argumentado por Buades, ao considerar os grupos de pixels mais semelhantes do conjunto do pixel alvo independente da localização, a diferença de posição no quadro deixa de ser um problema [21]. O NLM é continuamente melhorado ao otimizar a busca pelos grupos ou segmentos corretos, utilizando abordagens estatísticas no lugar da distância euclidiana, sem contudo perder a ideia original do algoritmo [22].

### 2.3.6 *Video Block-Matching 3D*

Inspirado pelo agrupamento não-local do NLM, o algoritmo *Block-Matching and 3D Filtering* (BM3D) proposto por Dabov et al. aprimora a estratégia de utilizar fragmentos similares para restaurar o pixel alvo, separando-os em grupos onde cada grupo possui estrutura local semelhante [23]. O método possui duas etapas. A primeira inclui a busca de pequenos fragmentos similares que são empilhados em um vetor tridimensional, onde é calculada uma transformação linear (como transformada discreta de Fourier, cosseno ou *wavelet*). É feita uma limiarização dos coeficientes obtidos e a transformação inversa é agregada em uma estimação da imagem filtrada.

A segunda etapa é semelhante à primeira, mas o agrupamento de fragmentos similares considera a imagem original e a saída da primeira etapa. No lugar da limiarização, é utilizado o filtro de Wiener [24] para produzir a estimativa final da restauração com uma média ponderada da estimativa dos grupos em relação a cada pixel alvo. O desempenho do BM3D depende linearmente do parâmetro que regula a intensidade da filtragem na

etapa de limiarização, devendo corresponder ao desvio padrão esperado do ruído. Assim como no NLM, o BM3D foi adaptado em uma versão *interframe* para vídeos denominada VBM3D [25].

## 2.4 Métricas de Qualidade

É importante que existam métricas que permitam quantificar a qualidade de um vídeo ou imagem, de forma que seja possível avaliar o desempenho de serviços automaticamente, como por exemplo em um sistema de transmissão de vídeos, ou ainda a eficiência de métodos para restauração e aprimoramento de imagens ou vídeos [26].

Na maioria dos casos, as aplicações de vídeo ou imagem são utilizadas por humanos, então são os próprios humanos que podem decidir com maior precisão o seu grau de qualidade. A avaliação realizada por pessoas é denominada métrica subjetiva, como por exemplo no caso do *Mean Opinion Score* (MOS), ou pontuação média das opiniões [27]. Porém, devido ao custo em tempo e dinheiro, é inviável que se utilize humanos em processos de controle de qualidade. Assim, se faz necessária a utilização de métricas objetivas que possam ser calculadas automaticamente por uma máquina, e que possuam alta correlação com as métricas subjetivas como o MOS [26].

As métricas objetivas são classificadas em três categorias [28]:

- Referência Completa (*Full Reference*): podem ser utilizadas se a versão original ou de referência do vídeo também está disponível.
- Referência Parcial ou Reduzida (*Reduced Reference*): quando se tem algumas informações ou parâmetros do vídeo original.
- Sem Referência (*No-Reference*): são utilizadas se apenas o vídeo distorcido ou processado está disponível para análise.

Este trabalho aborda algumas métricas de referência completa e sem referência, que são descritas nas próximas subseções.

### 2.4.1 Métricas de Referência Completa

As métricas de referência completa requerem além da imagem ou vídeo ruidosos ou processados, a presença da sua versão original para comparação, considerada como a versão “perfeita” ou não degradada.

#### 2.4.1.1 *Mean Squared Error*

O MSE calcula o erro quadrático médio entre dois conjuntos de valores. Em processamento de imagens, também é utilizada para comparar duas imagens ou quadros de

vídeo, assim a métrica mede o quão diferentes duas imagens são. Pode ser definida pela equação (2.15) [26]:

$$MSE = \frac{1}{N} \sum_{i=1}^N (X_i - Y_i)^2, \quad (2.15)$$

onde  $N$  é o número total de pixels da imagem ou quadro de vídeo, e  $X_i$  e  $Y_i$  são os  $i$ -ésimos pixels da versão original e processada, respectivamente. A métrica é adaptada para vídeo considerando os valores obtidos para cada quadro, logo, os vídeos devem possuir a mesma quantidade de quadros. Quanto mais alto o resultado da métrica MSE, maior a diferença entre os vídeos [26].

#### 2.4.1.2 *Peak Signal-To-Noise Ratio*

O PSNR ou “relação sinal-ruído de pico” define a similaridade entre dois sinais, estando relacionado ao MSE como pode ser observado pela fórmula (2.16), onde 255 representa a intensidade máxima de um pixel da imagem [29].

$$PSNR = 10 \cdot \log_{10} \frac{255^2}{MSE} \quad (2.16)$$

A avaliação com a PSNR é feita a partir do MSE resultante do vídeo. Quanto maior o valor da métrica PSNR, maior a “fidelidade” do vídeo em relação ao original. A unidade da PSNR é o decibel (dB), cujo valor para vídeos muito próximos da referência varia entre 30 e 40 dB [27]. As métricas MSE e PSNR ainda são muito utilizadas devido à baixa complexidade, mas não correspondem bem com as métricas subjetivas, já que representam respectivamente a quantidade de diferença e similaridade entre duas imagens ou vídeos de forma “puramente matemática”, sem incorporar elementos da qualidade percebida pelo sistema visual humano [26]. Uma utilização mais adequada destas métricas é quando o conteúdo não se altera, como na comparação de técnicas de compressão com perda de informação [30].

#### 2.4.1.3 *Structural Similarity*

A métrica SSIM proposta por Wang et al. (2004) é baseada na ideia de que imagens naturais são altamente estruturadas. Isto é, os sinais existentes em uma imagem digital possuem fortes relações entre si, contendo informações sobre as estruturas dos objetos na cena [31]. O algoritmo mede a similaridade estrutural de uma imagem  $y$  em relação à referência  $x$  a partir de 3 características: a luminância, o contraste e a estrutura [26]. As três componentes são combinadas na fórmula dada pela equação (2.17) [31], onde as constantes são calculadas em relação à amplitude do valor dos pixels.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (2.17)$$

O resultado da métrica SSIM pode variar entre 0 e 1, onde um valor mais alto indica maior similaridade. A utilização da métrica em vídeos como uma adaptação *intraframe* não é capaz de detectar artefatos temporais [32], mas ao contrário do MSE e PSNR, a SSIM possui maior correlação com as métricas subjetivas [27].

#### 2.4.1.4 Video Quality Metric

O algoritmo VQM é uma métrica específica para vídeos, proposta por Wolf e Pinson (1999) da *National Telecommunications and Information Administration* (NTIA), que consiste em dividir o vídeo processado e o de referência em vários segmentos denominados blocos espaço-temporais, e para cada um deles mensurar e comparar diversas características visuais como distorções de cor, borrões, *blocking*, movimentação irregular e ruídos globais [33]. A técnica pode ser dividida nos seguintes estágios [26]:

- Calibração: é a fase de preparação dos vídeos para a extração de características, onde deslocamentos espaciais e temporais são corrigidos e diferenças de brilho e contraste são compensadas.
- Extração de características de qualidade: nesta etapa um conjunto de atributos relacionados à percepção da crominância e intensidade no espaço e tempo é extraída dos segmentos de cada vídeo. O cálculo das características é feito a partir de filtros de realce seguidos de limiarização.
- Estimação dos parâmetros de qualidade: um conjunto de parâmetros é calculado pela comparação entre as características extraídas do vídeo processado e de referência.
- Estimação da qualidade: o resultado final da métrica é calculado usando uma combinação linear de todos os parâmetros de qualidade.

Assim como na SSIM, o resultado da VQM varia entre 0 e 1, onde um valor mais alto indica qualidade superior. O algoritmo VQM possui a maior complexidade entre as métricas objetivas de qualidade, mas também apresenta a melhor correspondência com as métricas subjetivas, tendo obtido um coeficiente de correlação de 95% em testes realizados por Wolf e Pinson [27, 29].

#### 2.4.2 Métricas Sem Referência

As métricas sem referência avaliam características ou artefatos específicos da imagem ou vídeo alvo, sem utilizar uma versão de referência para comparação [26]. A seguir são descritas as principais métricas sem referência usadas pelo trabalho na extração de atributos dos vídeos.

### 2.4.2.1 Rank-Ordered Absolute Differences

A métrica *Rank-Ordered Absolute Differences* (ROAD) calcula a proximidade do valor de um pixel para os seus quatro vizinhos mais similares. Ela se baseia na ideia de que impulsos variam muito em intensidade em relação à maioria ou todos os pixels vizinhos, enquanto os pixels do conteúdo deveriam possuir no mínimo metade dos pixels vizinhos com intensidade semelhante, mesmo pixels de borda [34].

No exemplo da equação 2.18 é mostrado o procedimento para calcular o ROAD do pixel central de uma janela 3x3 na primeira matriz. A segunda matriz mostra as diferenças absolutas para o pixel alvo, onde os quatro menores valores para somatória do resultado são destacados. É possível observar que a coluna esquerda representando uma borda possui pixels de intensidade semelhante, resultando em um ROAD mais baixo.

$$\begin{bmatrix} 213 & 171 & 88 \\ 216 & 186 & 107 \\ 218 & 202 & 139 \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{27} & \mathbf{15} & 98 \\ \mathbf{30} & - & 79 \\ 32 & \mathbf{16} & 47 \end{bmatrix} \quad (2.18)$$

$$ROAD = \sum_{i=1}^4 r_i = 15 + 16 + 27 + 30 = 88$$

O valor final da métrica é considerado como a média de todos os ROAD calculados. A tendência em uma imagem ou vídeo ruidoso é a ocorrência de vários pixels com poucos ou nenhum vizinho semelhante, aumentando o valor resultante. Assim, a métrica ROAD é utilizada como um bom indicador de ruído, em especial o ruído impulsivo [34].

### 2.4.2.2 Brightness Flicking Metric

A métrica BFM ou “oscilação do brilho” é um algoritmo simples que calcula a diferença absoluta da intensidade média do canal de luminância entre dois quadros adjacentes de vídeo, podendo ser definida pela equação (2.19) [35]:

$$BFM_i = ||\bar{L}_i - \bar{L}_{i+1}||, \quad (2.19)$$

onde  $\bar{L}_i$  é o valor médio do canal de luminância de um quadro  $i$  do vídeo, e  $\bar{L}_{i+1}$  o valor do próximo quadro. O resultado final da métrica é considerado como a média de todos os  $BFM_i$  calculados. A presença de ruído tende a aumentar a oscilação das intensidades dos quadros.

### 2.4.2.3 Blurriness

As métricas de *blurriness* ou “borramento” são baseadas na ideia de que borrões tornam as bordas maiores ou menos nítidas. Assim, os algoritmos calculam o borramento estimando o tamanho e intensidade das bordas [26].

A primeira etapa do algoritmo consiste em encontrar bordas nítidas utilizando um detector de bordas como o Canny ou Sobel, que são filtros passa-alta, conforme descrito na seção 2.3. Para cada pixel  $(i, j)$  de uma imagem, a saída do detector de bordas fornece a magnitude  $M(i, j)$  e a orientação  $O(i, j)$ . Para selecionar apenas os pixels relacionados às bordas mais nítidas, utiliza-se um limiar padrão, como por exemplo selecionando os pixels onde  $M(i, j) > 25$ . Dessa forma, pode-se calcular a métrica de borramento pela equação (2.20) [36]:

$$Blurriness = \frac{1}{L} \sum_{i=0}^N \sum_{j=0}^M W(i, j), \quad (2.20)$$

onde  $L$  é o número de pixels selecionados das bordas nítidas, e  $W$  é uma função calculada pela distância entre dois extremos locais  $P_1$  e  $P_2$  de uma borda, que indica o seu tamanho. Assim, o resultado será a média de tamanho das bordas de maior magnitude em relação a cada pixel contido em uma borda. Quanto mais baixo o valor da métrica, menos borrada e mais nítida é a imagem, sendo reduzido na presença de ruídos. Em vídeos, o resultado é a média do valor obtido pela métrica em cada quadro [36].

### 3 TRABALHOS RELACIONADOS

Um dos trabalhos que consideram a estrutura de ruído presente em vídeos aponta que enquanto grande parte do desenvolvimento de algoritmos de redução de ruído assume uma generalização para a distribuição de ruído Gaussiano, não apenas vídeos apresentam mais ruído que imagens, mas também que o ruído em vídeos frequentemente possui diferentes características e a eficiência da redução de ruído é diretamente relacionada ao modelo de ruído esperado pela técnica de filtragem. A partir dessa premissa, Ji et al. [5] mostra que um sistema eficiente para redução de ruído real deve lidar com a maioria das distribuições de ruído que ocorrem na aquisição de vídeos. Para avaliar o seu algoritmo proposto, eles conduzem testes que consistem de vídeos degradados por uma mistura de ruído Gaussiano, *poisson* e impulsivo, comparando com outros algoritmos de redução de ruído como o VBM3D através da métrica PSNR, mostrando que algoritmos projetados para a remoção do ruído Gaussiano não são tão eficientes quando outras distribuições de ruído estão presentes.

A redução de ruído em imagens para o ruído Gaussiano, impulsivo e gama é avaliada em [37] por métricas de referência completa incluindo PSNR e SSIM, métrica empregada em vídeos como um método *intraframe* [32]. O filtro bilateral obteve melhor desempenho nos resultados para o ruído gama, enquanto para o ruído impulsivo o filtro de mediana foi claramente superior. A maioria dos trabalhos relacionados mostram que o filtro de mediana e suas variantes são ideais para a remoção de ruído impulsivo, pois sua resposta a impulsos é sempre zero [38, 39], enquanto métodos como o filtro bilateral e anisotrópico falham na sua remoção já que tratam o ruído como bordas nítidas [13]. Em outro trabalho [40], a redução de ruído em imagens e vídeos com variantes do NLM e VBM3D também é avaliada pela métrica SSIM. Sutour et al. compara a remoção dos ruídos gama e *poisson* além do Gaussiano, mas mesmo a sua técnica variante do NLM não alcançou melhores resultados que o VBM3D devido à suavização excessiva do NLM, onde o ruído foi removido às custas de grande parte da qualidade original.

Os trabalhos na área de redução de ruído indicam a necessidade de avaliar respostas de filtragem buscando cenários mais diversificados, onde outros tipos de ruído além do Gaussiano prevalecem. Outro ponto é o uso frequente de métricas de imagens como a PSNR e SSIM, com baixa utilização das métricas que possuem boa correlação com a percepção humana e que permitem avaliar vídeos, onde a qualidade temporal também é levada em consideração. Um exemplo de tal métrica é a VQM (*Video Quality Metric*), descrita na seção 2.4.1.4. Recentemente, Arias e Felinto [41] conduziram experimentos de redução de ruído em vídeos empregando as métricas SSIM e VQM, mostrando diferenças significativas na avaliação da filtragem *interframe*, indicando que apesar de convergirem

na avaliação objetiva, cada métrica possui uma sensibilidade diferente às distorções, aos efeitos de suavização causados pelos algoritmos de restauração e ao domínio temporal. Além disso, as oscilações no desempenho da filtragem em relação a cada tipo e nível de ruído justificam uma abordagem de adaptação da técnica de restauração de acordo com o modelo de ruído presente.

Para solucionar o problema de detecção e estimação de ruído sem conhecimento prévio do mecanismo de degradação, os métodos iniciais consistiam em análise de histograma baseada em uma perspectiva global, permitindo estimar a ocorrência de um tipo de ruído [2], mas é uma abordagem limitada com suposições inválidas na maioria das aplicações com conteúdo não homogêneo [42]. Esse tipo de técnica foi aprimorada com abordagens de seleção de áreas homogêneas para análise de histogramas locais [43]. Outras propostas utilizaram técnicas de detecção de bordas [44] ou atributos estatísticos [45] para isolar o ruído e identificá-lo entre os principais tipos.

Em relação ao ruído impulsivo, há uma variedade de trabalhos para detecção de cada pixel corrompido com aplicação de uma versão modificada do filtro de mediana. Eles geralmente são baseados nas diferenças absolutas entre valores ordenados de conjuntos de pixels, como o *rank-ordered absolute differences* (ROAD) [34]. No caso do ruído impulsivo, a classificação de cada pixel pode ser usada diretamente para filtragem com um algoritmo que considera apenas pixels sem ruído para restauração, utilizando os valores médios ou medianos dependendo da quantidade de pixels ruidosos em cada janela [46].

Com o advento de métodos com aprendizado de máquina, as propostas passaram a utilizar modelos treinados para identificar classes de ruído distintas. Diferentes trabalhos incluem a utilização de uma árvore de decisão C4.5 com dez atributos estatísticos comuns tais como média, desvio padrão e entropia dos pixels para detecção de ruído impulsivo, aleatório e a ausência de ruído [47]; a mensuração de atributos de assimetria e curtose com uma rede neural probabilística para classificação entre ruído impulsivo, Gaussiano e não-Gaussiano [48]; e entre ruído Gaussiano, *speckle*, impulsivo e não-Gaussiano [49] por outro classificador com redes neurais, obtendo os melhores resultados apenas nas amostras com mais regiões homogêneas e baixo nível de ruído. Outra abordagem específica para detecção de ruído com seleção de blocos envolveu clusterização com uso de atributos morfológicos [50], e aplicação do algoritmo de otimização por enxame de partículas para detecção de ruído impulsivo, Gaussiano ou uma combinação de ambos [51].

Grande parte das propostas para detecção de ruído executa classificação binária incluindo o ruído Gaussiano ou impulsivo, apesar de reconhecerem a possibilidade de expandir o escopo do modelo para inclusão de outros tipos, a classificação multiclasse de ruído é pouco considerada. Além disso, a maioria dos trabalhos considera variações de classes de ruído aditivo que são independentes do sinal original, em oposição aos tipos de ruído multiplicativos que são correlacionados ao conteúdo como o ruído *speckle*.

Na estimação do nível ou intensidade do ruído, o modelo mais considerado é o ruído aditivo Gaussiano. Um grande número de propostas requerem a existência de áreas homogêneas para minimizar o erro do nível estimado, uma vez que quando o conteúdo original é uniforme, a variância da região é equivalente à variância do ruído [52]. Similamente à detecção do tipo de ruído, a maioria das técnicas com essa suposição inclui uma etapa de seleção de áreas homogêneas, buscando a separação do sinal legítimo para estimação do ruído [42]. Uma das técnicas envolve a utilização de um filtro passa-alta como o Laplaciano para isolar os componentes de alta frequência contendo bordas, detalhes e ruído, seguido da remoção das bordas para obter a estimação do ruído a partir do conjunto resultante [53]. Outro método conhecido utiliza transformada *wavelet* (WT), onde a estimação do ruído é realizada partir do desvio absoluto mediano dos coeficientes no melhor nível de decomposição utilizando *wavelets* de Daubechies [54], o que geralmente resulta em superestimações pois os coeficientes também são afetados pela estrutura do conteúdo legítimo.

Como a presença de regiões homogêneas não é uma suposição consistente na maioria das aplicações, outros algoritmos foram propostos, em especial métodos baseados em análise de componentes principais (PCA), que utilizam o menor autovalor da matriz de covariância como a estimação do ruído [55]. Neste caso, a técnica aproveita-se da premissa de que as estruturas de textura do conteúdo na imagem são redundantes e autocorrelacionadas, enquanto o ruído é independente. A estimação com transformada discreta de cosseno (DCT) é outro método de estimação que se mostra eficaz sem a presença de regiões homogêneas, onde é explorada a premissa das estruturas da imagem ocuparem as baixas frequências, enquanto as altas frequências da transformação contêm apenas o ruído [56].

## 4 PROCEDIMENTOS METODOLÓGICOS

Neste capítulo, são descritos os materiais e procedimentos empregados para obtenção dos resultados no capítulo 5. A Figura 1 ilustra o funcionamento do sistema com cada uma das partes descritas neste capítulo: a avaliação das técnicas de filtragem em relação a diferentes tipos e níveis de ruído pela métrica VQM; a detecção do tipo e estimação do nível de ruído com algoritmos de aprendizado supervisionado; e a partir dos dados de restauração e o resultado da detecção e estimação, a seleção da técnica e parâmetros de redução de ruído mais adequados de acordo com a maior pontuação obtida para o respectivo tipo e nível de ruído resultantes do modelo preditivo.

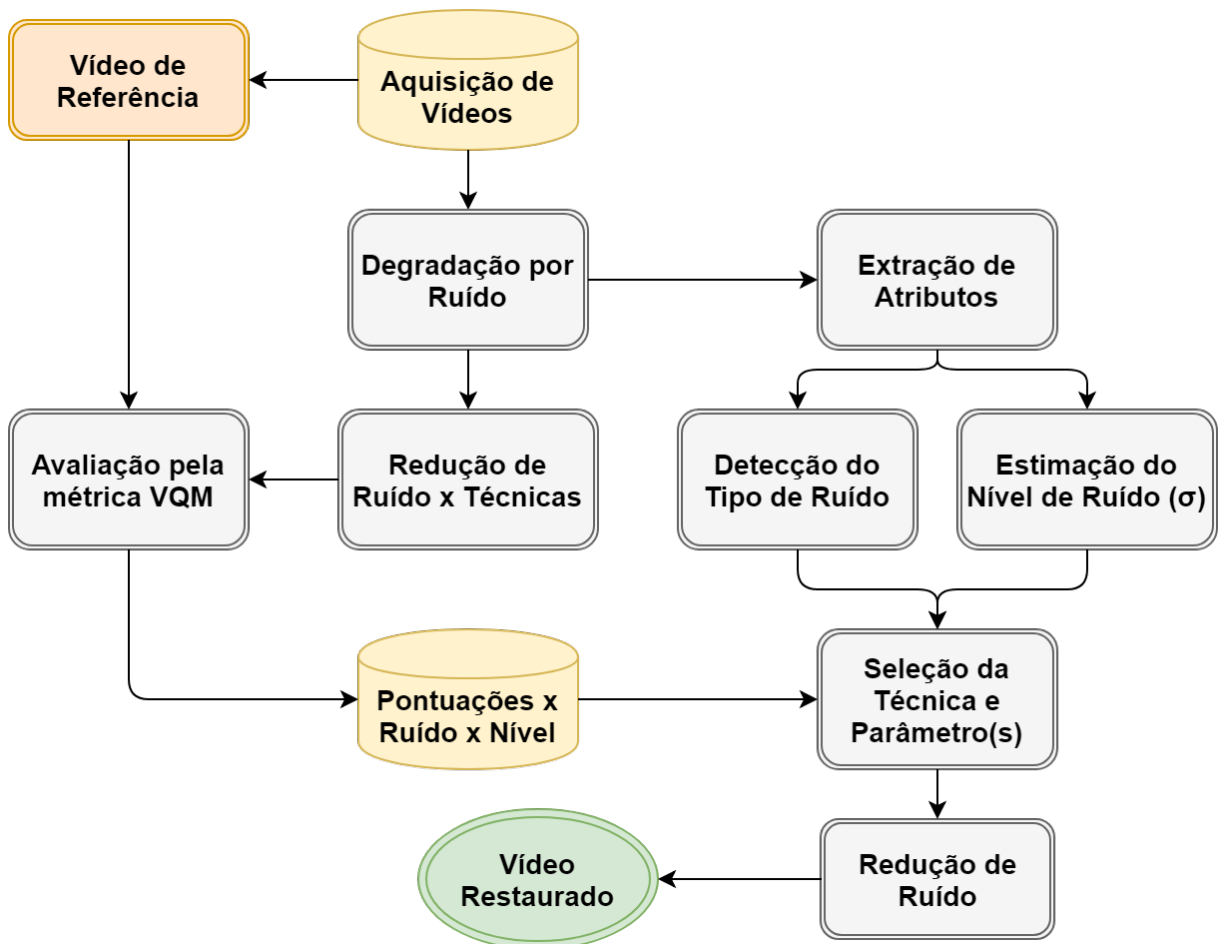


Figura 1 – Fluxograma do sistema adaptativo de detecção e remoção de ruídos.

### 4.1 Aquisição

Foram utilizadas 12 bases de dados na execução dos experimentos, totalizando 2000 vídeos de referência distintos de cenários reais com alta variação de conteúdo:

- *Video Quality Experts Group* (VQEG) ANSI e vídeos de referência da *Phase I* [57]: 336 amostras de vídeo em definição HD (1920x1080 pixels) na codificação H.264 e MPEG-2 sem erros de transmissão;
- Brandao et al. [58]: 56 vídeos de resolução 720x480 codificados em H.264 sem erros de transmissão;
- *Influence Content* [59]: 71 vídeos de resolução padrão (640x480 pixels) com codificação H.264/SVC sem erros de transmissão, sendo consideradas as versões de referência identificadas com a condição zero de significado “referência VGA” (sem degradação);
- IVP [60]: 138 vídeos em alta definição (resolução de 1920x1088 pixels) no modo progressivo (HDp) e formato 4:2:0;
- *LIVE Flicker* [61] e *LIVE Mobile* [62]: total de 150 vídeos com resolução 768x432 pixels codificados em H.264 e formato 4:2:0 no modo progressivo;
- Dorr et al. [63]: 180 vídeos em resolução 1280x720 codificados em MPEG-2;
- MMSys [64]: 240 vídeos em resolução padrão (640x480 pixels) codificados em MPEG-4 no formato 4:2:0;
- *Privacy evaluation videos* (PEV) [65]: 21 vídeos com resolução *full* HD (1920x1080) codificados no formato MPEG-2;
- *Video Trace Library* [66]: base com 26 vídeos em formato QCIF (176x144) e CIF (352x288) codificados em YUV no formato 4:2:0;
- YouTube-8M [67]: 782 vídeos selecionados de 32 categorias distintas de conteúdo com resolução 640x360 codificados em H.264/MPEG-4 AVC no formato 4:2:0, cuja lista de vídeos está disponível em [68].

A Figura 2 mostra quadros de vídeos selecionados aleatoriamente das primeiras seis bases de dados listadas; a Figura 3 ilustra mais cinco bases de vídeos, enquanto a Figura 4 contém vídeos da base do YouTube-8M. A maioria das bases de dados contém informações e análise subjetiva sobre suas características espaciais e temporais, mas como pode ser observado, o propósito do seu uso neste trabalho é a variação máxima do conteúdo original dos vídeos de forma que os modelos de detecção e estimação de ruído possam isolar a informação do ruído com a maior generalização possível.

A base de dados com os 2000 vídeos de referência foi aleatoriamente distribuída entre quatro classes distintas: 500 vídeos foram mantidos inalterados, enquanto os outros 1500 foram corrompidos com ruído Gaussiano, impulsivo e *speckle*. Um ruído definido por uma função de distribuição aleatória pode variar em sua intensidade, geralmente indicada



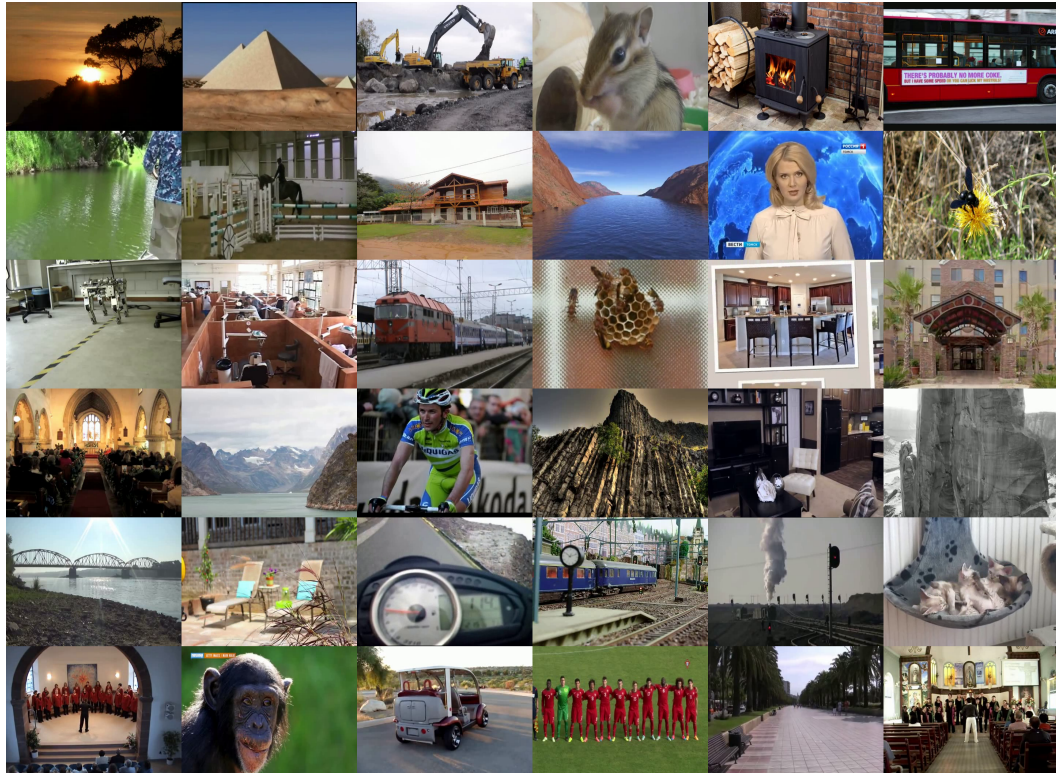


Figura 4 – Base selecionada pelo YouTube-8M contendo 32 categorias de vídeos.

pelo desvio padrão ( $\sigma$ ) [9], assim, para cada classe, dez níveis de ruído foram considerados em relação ao valor de  $\sigma$ : 10, 15, 20, 25, 30, 35, 40, 45, 50 e 55. O desvio padrão do ruído sempre representa o desvio padrão do componente aditivo do ruído. Para o ruído *speckle* que é um ruído multiplicativo, mesmo sendo correlacionado com o vídeo, o nível real do ruído é estimado a partir do componente aditivo resultante em  $i = i + (i * n)$  onde  $i$  é uma imagem ou quadro de vídeo e  $n$  o componente de degradação.

Utilizando o modelo de cores HSV descrito na seção 2.1.1, o ruído foi aplicado em cada canal dos quadros de vídeo. Deve ser destacado que é utilizado o mesmo tipo e nível de ruído na totalidade dos quadros de um vídeo. Em aplicações onde são verificadas degradações distintas em diferentes partes de um mesmo vídeo, uma alternativa para uso do sistema deste trabalho seria a divisão dos quadros e a consideração de cada fragmento como um vídeo independente.

As Figuras 5, 6 e 7 apresentam amostras de vídeos para cada classe de ruído com os níveis de ruído associados ao desvio padrão de 20, 30, 40 e 50 respectivamente.

## 4.2 Avaliação das técnicas de redução de ruído

A maioria da implementação dos testes de processamento de vídeo foi executada na linguagem C++ utilizando a biblioteca de funções do OpenCV<sup>1</sup>, que possui várias

<sup>1</sup> <<http://opencv.org>>

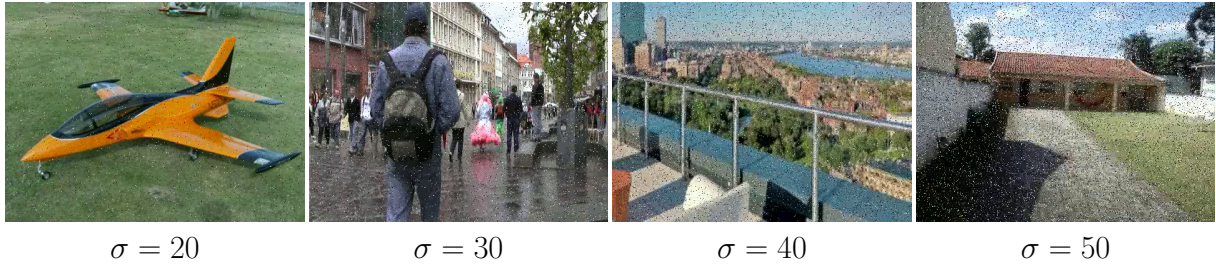


Figura 5 – Vídeos corrompidos com diferentes níveis de ruído impulsivo.

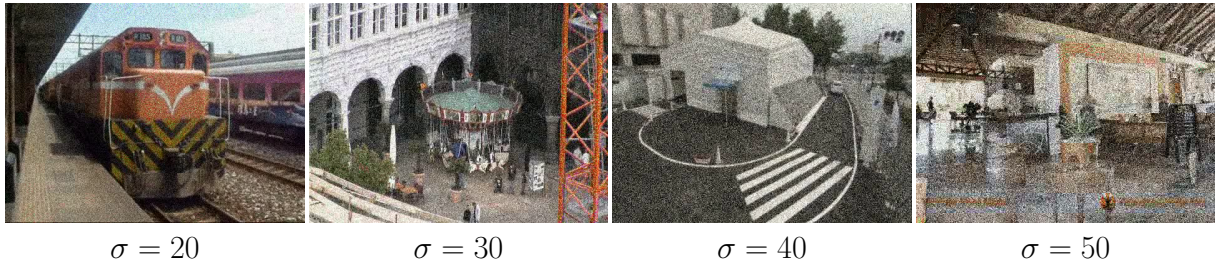


Figura 6 – Vídeos corrompidos com diferentes níveis de ruído Gaussiano.

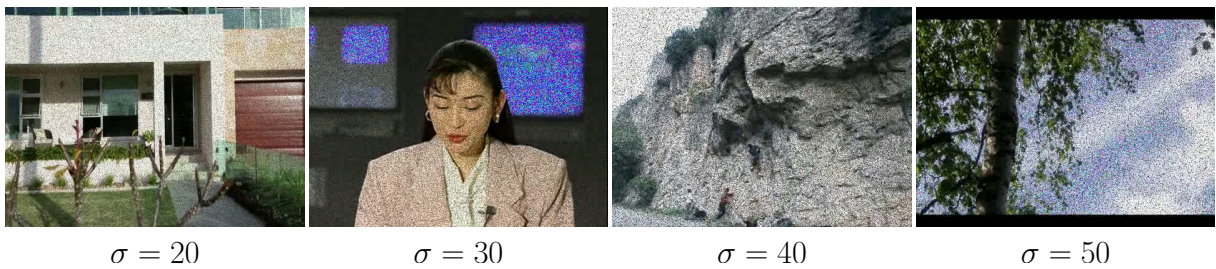


Figura 7 – Vídeos corrompidos com diferentes níveis de ruído *speckle*.

ferramentas para manipulação de imagem e vídeo. A Tabela 1 apresenta os algoritmos de redução de ruído que foram descritos na seção 2.3 e os respectivos parâmetros de regulação da filtragem considerados nos experimentos. No caso do filtro de mediana, o parâmetro utilizado determina o número de iterações aplicadas com tamanho fixo de janela para menor borramento. Para os filtros NLM e VBM3D, é empregada a versão *interframe* dos algoritmos em que cada quadro de vídeo é processado em conjunto com dois quadros vizinhos.

A Figura 8 ilustra o procedimento de avaliação das técnicas de redução de ruído. Para a avaliação do desempenho, cada algoritmo de filtragem é aplicado em todos os vídeos da base de dados, onde cada vídeo processado resultante é avaliado objetivamente pela métrica VQM, que requer o vídeo original para comparação com a versão degradada ou processada. Como descrito na seção 2.4.1.4, as pontuações da métrica VQM podem variar entre 0 e 1, onde valores mais altos indicam qualidade visual superior. Com exceção do VBM3D cuja versão utilizada possui o nível de ruído como parâmetro direto de entrada, os parâmetros foram iterados usando um passo fixo buscando obter a maior pontuação

Tabela 1 – Técnicas de redução de ruído e respectivos parâmetros considerados nos experimentos.

<b>Filtro</b>	<b>Parâmetros fixados</b>	<b>Parâmetros variáveis</b>
Gaussiano	$ksize = 3$	$\sigma_X$
Mediana	$ksize = 3$	$niters$
Bilateral	—	$\sigma_t, \sigma_s$
Anisotrópico	$niters = 15$	$alpha$
NLM	$templateSize = 7, searchSize = 42$	$h$
VBM3D	$templateSize = 7, searchSize = 42$	$\sigma$

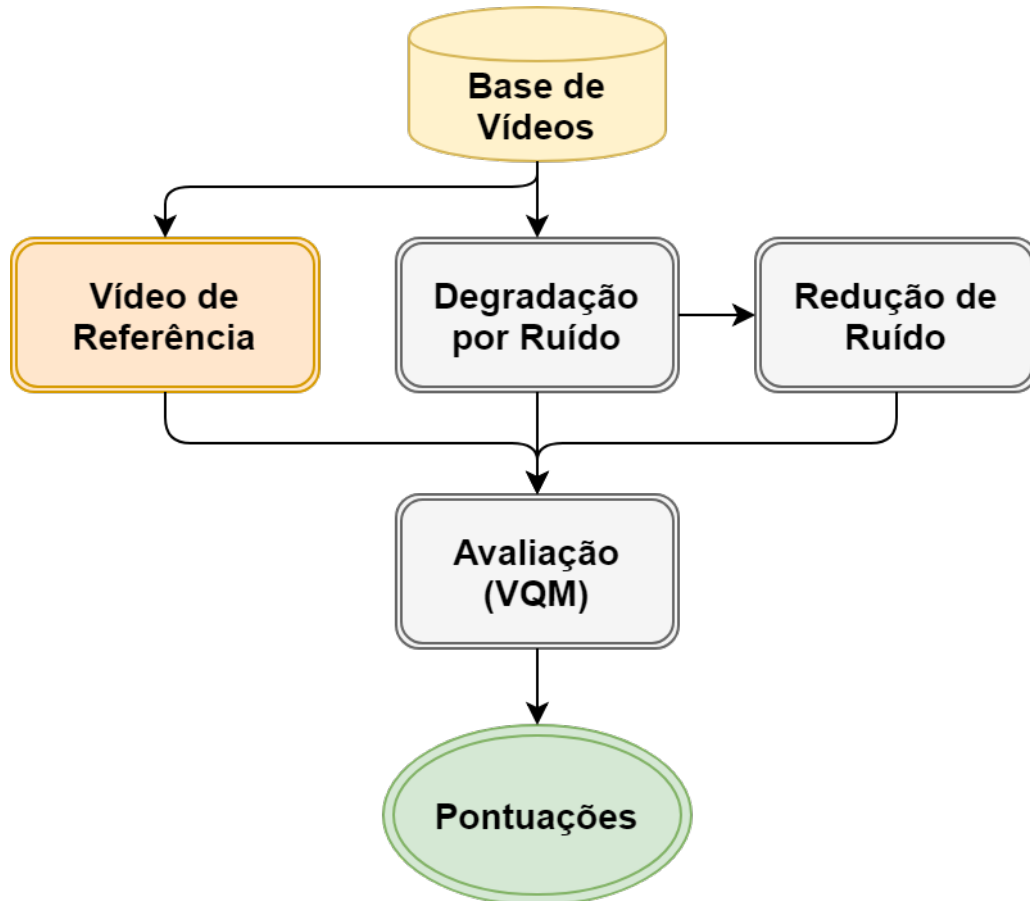


Figura 8 – Fluxograma para avaliação dos algoritmos de redução de ruído.

pela métrica VQM, objetivando avaliar a capacidade máxima de restauração de cada técnica para determinada instância. O ciclo mais externo é a mudança de conteúdo com diferentes amostras de vídeo. Para um resultado generalizado, é considerado o valor médio de pontuação dos diferentes vídeos agrupados pelo mesmo tipo e nível de ruído.

A partir das pontuações obtidas nos experimentos, é construída uma tabela associando para cada tipo e nível de ruído a técnica de filtragem com seu respectivo parâmetro

que obteve a maior pontuação. Dessa forma, com a detecção de determinado tipo de ruído  $r$  e estimação de um nível de ruído  $n$ , a tabela permite selecionar a técnica e o parâmetro associado que obteve a maior pontuação VQM para o ruído  $r$  no nível de ruído mais próximo do valor  $n$  estimado.

### 4.3 Extração e seleção de atributos

Para treinar o modelo de classificação e regressão de ruído, foram considerados diversos atributos relacionados da literatura de processamento de imagem e vídeo. A seleção de atributos foi executada em duas etapas. Na primeira, um conjunto maior foi filtrado através da seleção de atributos por correlação (*correlation feature selection*), selecionando atributos de alta correlação com a predição enquanto não correlacionados entre si [69]. Adicionalmente, é realizada a seleção de atributos para cada algoritmo de aprendizado supervisionado considerado na seção 4.5 com eliminação recursiva de atributos utilizando a validação cruzada com 10 dobras. Além disso, foi executado antecipadamente o escalonamento e normalização dos atributos de acordo com o valor mínimo e máximo em cada atributo para otimizar a convergência dos métodos.

A Tabela 2 apresenta os atributos selecionados na primeira etapa, ordenados de acordo com as respectivas importâncias obtidas na classificação pelo *Random Forest* (RF) [70], que utiliza o índice Gini para maximizar a redução da impureza ao selecionar a melhor divisão do conjunto de dados. Os atributos incluem medidas estatísticas extraídas de cada amostra como média, desvio padrão, entropia, curtose e assimetria do histograma dos níveis de cinza, desvio absoluto mediano (MAD) [71], a métrica de borramento de vídeo (seção 2.4.2.3), blocagem (*blockiness*) [72], ROAD (seção 2.4.2.1) e a métrica de oscilação do brilho (BFM) (seção 2.4.2.2). Foram descartados outros atributos com baixa importância mensurada como mediana, valor máximo e mínimo dos níveis de cinza, a métrica de *ringing* e a redundante ROLD (*rank-ordered logarithmic differences*) [73] com funcionamento similar à métrica ROAD mas calculada em uma escala logarítmica.

Métricas como borramento e blocagem são capazes de realçar as características de ruído em vídeos [36], enquanto o ROAD possui a maior capacidade de discriminação em relação ao ruído, baseado na suposição de que a variância local é alterada com a presença de ruído e a maioria das regiões ruidosas apresentam maior diferença de valores entre o pixel mediano em comparação a regiões regulares. Pode ser observado que os atributos não são computacionalmente complexos, permitindo a implementação em tempo real.

### 4.4 Pré-processamento e fusão de atributos

Em trabalhos relacionados descritos no capítulo 3, existem abordagens de realizar a filtragem seguida de diferença para análise do ruído isolado. Neste trabalho, o método é

Tabela 2 – Atributos selecionados na primeira fase e as respectivas importâncias.

Atributo	Importância (RF)	Descrição
ROAD	0.290	<i>Rank-ordered absolute differences</i> [34]
<i>Blurriness</i>	0.201	Métrica de borramento [36]
BFM	0.104	Métrica de oscilação do brilho [35]
Média	0.079	Média dos níveis de cinza
Curtose	0.075	Curtose do histograma
Assimetria	0.057	Assimetria do histograma
Blocagem	0.055	Métrica <i>blockiness</i> [72]
Entropia	0.055	Entropia dos níveis de cinza
Desvio padrão	0.054	Desvio padrão dos níveis de cinza
MAD	0.030	Desvio absoluto mediano [71]

adaptado para o contexto de aprendizado de máquina com aplicação de filtragem seguida de nova extração de atributos para fusão na entrada dos modelos de aprendizado supervisionado, como ilustrado na Figura 9. Este método é similar à análise de ruído “isolado” após a separação por segmentação, mas utiliza os atributos em conjunto com a extração original, permitindo a análise dos dados de vídeo por duas perspectivas distintas: antes e após a filtragem.

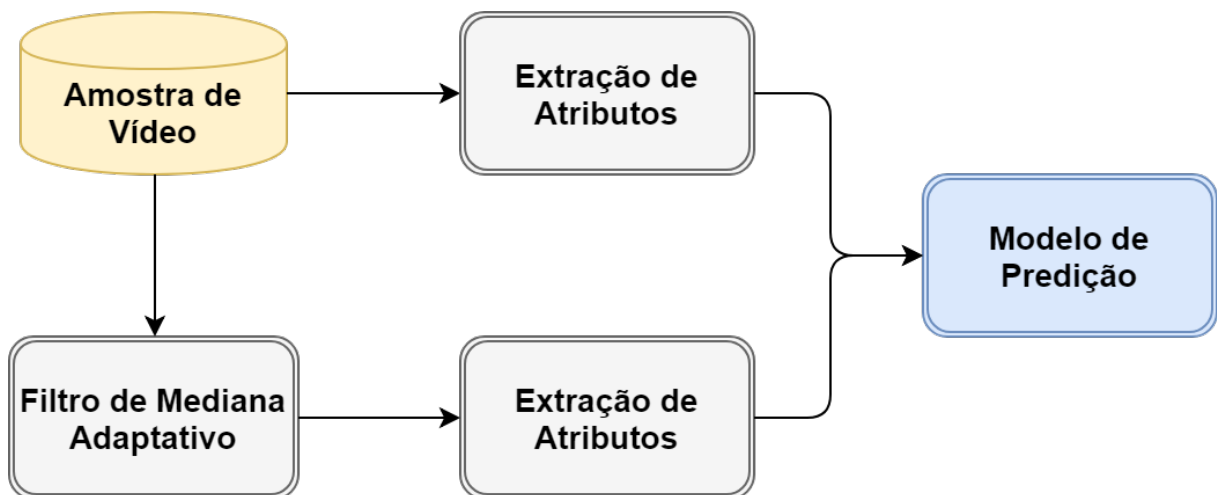


Figura 9 – Abordagem de pré-processamento do vídeo com filtragem e fusão de atributos.

Para isso, o pré-processamento é realizado com o filtro de mediana adaptativo descrito em [74]. Esta técnica aprimora o filtro de mediana comum com uma abordagem de filtragem dos pixels medianos das janelas sem a inclusão de valores extremos de 0 e 255, com maior preservação do conteúdo que o filtro de mediana comum. Apesar de não-

linear, o filtro de mediana comum ainda causa um nível de borramento, enquanto a versão adaptativa só remove pixels extremos. A utilização de outros filtros no pré-processamento não mostrou uma melhoria tão expressiva na predição, além disso, ao contrário do filtro de mediana eles são bastante sensíveis à regulação do parâmetro, tornando inviável extrair múltiplos conjuntos de atributos para a mesma amostra na detecção e estimação inicial do ruído.

Como o filtro de mediana e suas variantes representam a técnica ideal para remoção do ruído impulsivo ao eliminar completamente os valores extremos discrepantes, o resultado da filtragem nas amostras com ruído impulsivo é mais significativo do que nas amostras sem a presença de ruído ou com outros tipos de ruído, sendo verificável nos valores dos atributos após a filtragem e tornando amostras ambíguas mais facilmente classificáveis com as informações concatenadas para os modelos preditivos. A diferença nos resultados entre as abordagens dos modelos com e sem pré-processamento por filtragem e fusão de atributos é mostrada no capítulo 5.

## 4.5 Aprendizado supervisionado

Os testes de detecção e estimação de ruído realizam a comparação de seis diferentes algoritmos de aprendizado supervisionado: K vizinhos mais próximos (K-NN), árvore de decisão C4.5 [75], máquina de vetor de suporte (SVM) [76], rede neural perceptron multicamadas (MLP) [77], *Random Forest* (RF) [70] e *Gradient Boosting* (GBoost) [78]. O desempenho desses algoritmos varia de acordo com o ajuste dos hiperparâmetros, especialmente a SVM com a escolha do tipo do *kernel*, a constante de regularização  $C$  e o hiperparâmetro  $\gamma$ , assim eles foram testados a partir de um conjunto de parâmetros pré-definidos com busca em grade (*grid search*). Para implementação, foi utilizada a ferramenta do Weka<sup>2</sup> e a biblioteca scikit-learn<sup>3</sup> da linguagem Python.

Os experimentos empregaram a estratégia de validação cruzada com dez dobras (*10-fold Cross-Validation*) nas amostras, dessa forma, o resultado da predição computado para determinada amostra equivale ao resultado obtido em um dos dez particionamentos quando a respectiva amostra fazia parte do conjunto de teste e não do conjunto de treinamento, assim evitando o *overfitting* ou sobreajuste e aumentando a generalização do modelo resultante. Para avaliar a detecção de ruído são utilizadas algumas métricas de classificação como acurácia, precisão, *recall* ou taxa de verdadeiros positivos (TPR) e pontuação F1, enquanto na estimação com a regressão é empregado o erro quadrático médio (MSE) e o coeficiente de correlação de Pearson (PCC).

<sup>2</sup> <<https://www.cs.waikato.ac.nz/ml/weka/>>

<sup>3</sup> <<https://scikit-learn.org/stable/>>

## 5 RESULTADOS E DISCUSSÃO

Este capítulo apresenta e discute os resultados obtidos de acordo com a metodologia descrita no capítulo 4. Primeiramente, são avaliados os testes realizados com as técnicas de filtragem nos vídeos com ruído da base de dados em relação à pontuação pela métrica VQM em cada nível de ruído. Em seguida, são comparados os resultados de detecção de ruído pelo modelo proposto com os diferentes algoritmos de classificação e outras abordagens na literatura. A estimação do nível de ruído pelo modelo é avaliada pela diferença em relação ao nível real de ruído, pelas métricas de regressão e à remoção de ruído pela métrica VQM quando utilizado o nível estimado como entrada para o filtro VBM3D, em comparação à estimação por outros três métodos baseados em análise de componentes principais (PCA), transformada discreta de cosseno (DCT) e transformada *wavelet* (WT). Por fim, são analisados os resultados da seleção da técnica com o parâmetro associado de filtragem a partir da detecção e estimação realizadas pelo modelo proposto em relação aos métodos alternativos de estimação e à técnica de referência associada a cada tipo e nível de ruído, considerando tanto a capacidade de predição da técnica correta como o desempenho resultante da filtragem selecionada.

### 5.1 Avaliação das técnicas de filtragem de ruído

A Figura 10 apresenta as pontuações obtidas pela métrica VQM em relação à filtragem do ruído Gaussiano por cada técnica (filtro Gaussiano, de mediana, bilateral, anisotrópico, NLM e VBM3D) considerando os dez níveis de ruído na base de dados com desvio padrão ( $\sigma$ ) de 10 a 55. Cada valor da métrica representa a média das pontuações obtidas no processamento do conjunto de amostras com o respectivo nível de ruído. Como linha de base, a Figura 10 também contém as pontuações obtidas sem a aplicação de filtragem, indicando a avaliação pela métrica VQM da degradação do ruído em relação aos vídeos de referência sem qualquer tentativa de restauração.

Como pode ser observado a partir da Figura 10, nos menores níveis de ruído o filtro VBM3D obteve os melhores resultados, enquanto as pontuações das outras técnicas ainda se encontram relativamente altas com valores acima de 0.65. Para  $\sigma = 10$ , mesmo as amostras sem aplicação de filtragem apresentam uma pontuação média de 0.7, indicando que a aplicação dos filtros Gaussiano e de mediana é até mesmo prejudicial em baixas quantidades de ruído. Os outros níveis mostram que o filtro anisotrópico, NLM e VBM3D obtiveram os melhores resultados, enquanto as outras técnicas apresentam uma perda acelerada do desempenho seguida de estabilização da pontuação da métrica VQM com valores abaixo de 0.3 nos maiores níveis de ruído. O VBM3D produziu os melhores índices

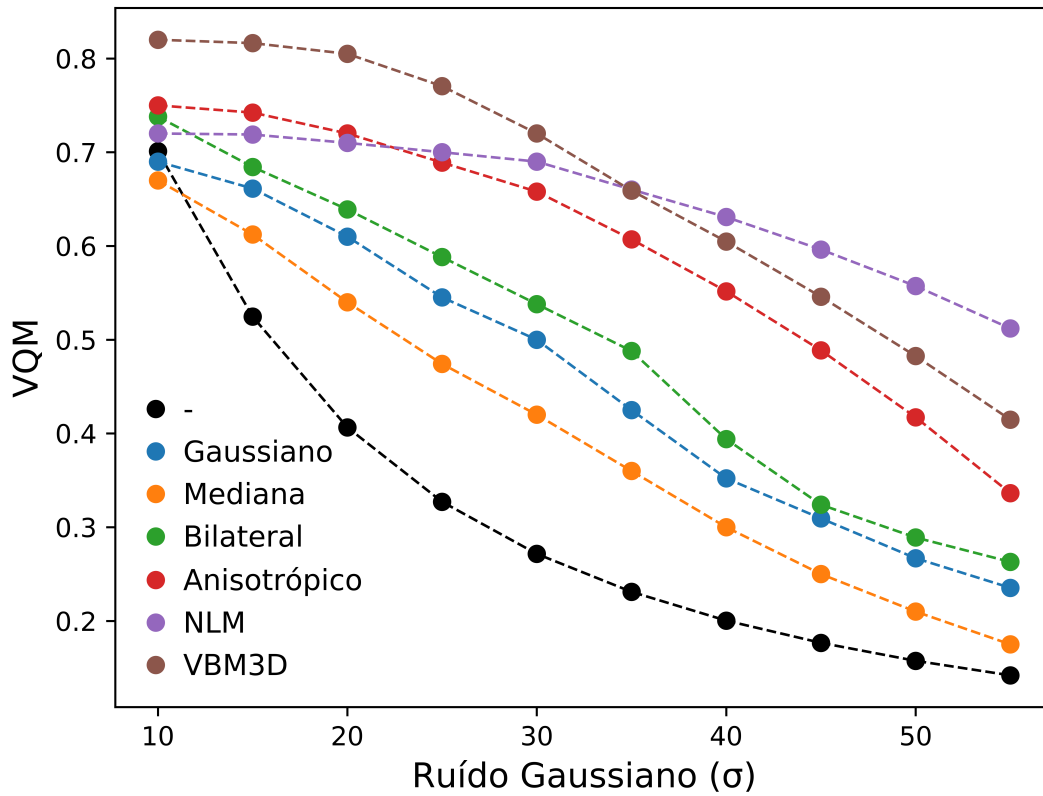


Figura 10 – Comparação dos algoritmos de filtragem para redução de ruído Gaussiano.

de restauração até  $\sigma = 35$ , enquanto o NLM foi mais eficiente nas maiores quantidades de ruído de  $\sigma = 40$  até  $\sigma = 55$ .

Como apontado em trabalhos relacionados no capítulo 3, para o ruído Gaussiano é esperado um desempenho superior das técnicas NLM e VBM3D, visto que esses algoritmos foram projetados especificamente para esse tipo de ruído. Apesar disso, o filtro bilateral e anisotrópico exibiram um desempenho mais alto que o NLM em alguns casos com menor nível de ruído quando um efeito intenso não era necessário, permitindo uma atenuação mais suave enquanto a técnica NLM não pôde executar menor remoção de ruído independentemente da regulação do parâmetro. Como descrito em trabalhos relacionados, o filtro NLM apresenta o fenômeno de “excesso de suavização” (*over-smoothing*) com maior intensidade em relação a outras técnicas, ou seja, o ruído tende a desaparecer mas a nitidez de bordas e detalhes da imagem também é deteriorada, efeito cuja compensação se mostra positiva somente nas maiores quantidades de ruído.

A Figura 11 mostra os resultados de processamento pelos diferentes algoritmos de filtragem de um quadro de vídeo deteriorado pelo ruído Gaussiano com  $\sigma = 40$ . O valor obtido pela métrica VQM na comparação do vídeo com a versão original de referência no primeiro quadro é mostrado entre parênteses. Pode ser observado que para essa instância

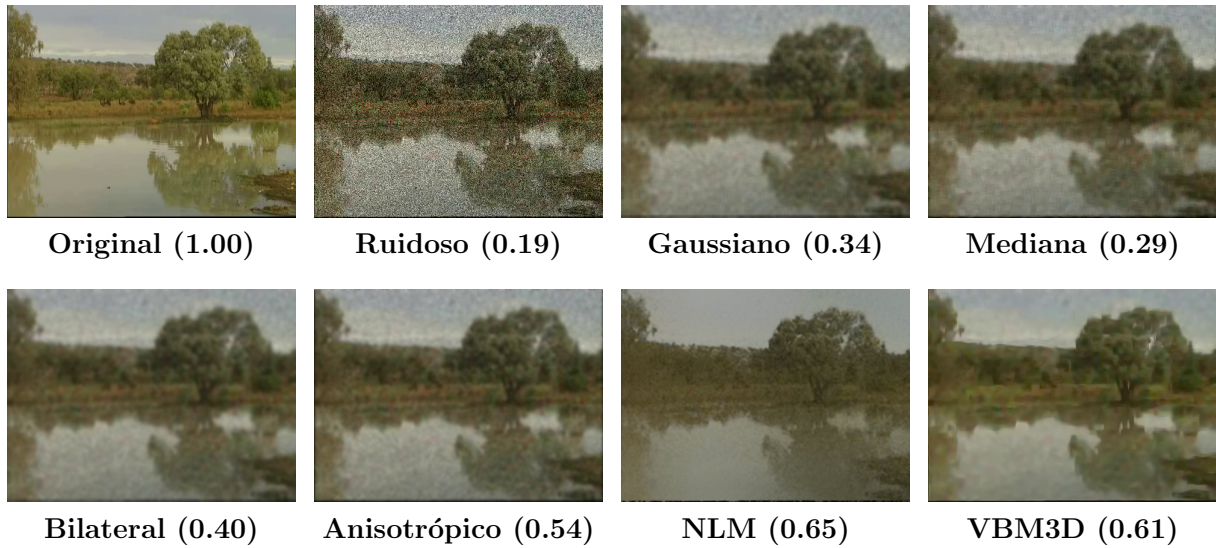


Figura 11 – Comparação visual da redução de ruído Gaussiano com  $\sigma = 40$ .

de ruído o NLM obteve os maiores resultados quantitativos, sendo também o respectivo vídeo filtrado o único onde não são perceptíveis manchas de ruído residuais na região do céu uniforme, mas ao mesmo tempo não foram preservadas outras características ainda visíveis na amostra filtrada pelo VBM3D que obteve a segunda maior pontuação.

A Figura 12 apresenta os resultados para a redução de ruído impulsivo, que também se comparam aos observados em trabalhos relacionados por outras métricas de qualidade como PSNR e SSIM. Pode ser verificado claramente que apenas o filtro de mediana foi capaz de remover o ruído impulsivo, com diminuição razoável das pontuações conforme o aumento do nível de ruído, enquanto o efeito positivo dos outros algoritmos foi bastante limitado na recuperação da qualidade dos vídeos. Como descrito nos trabalhos relacionados, devido às características desse tipo de ruído, técnicas como o filtro bilateral e anisotrópico não são efetivas pois tratam os pixels com ruído como bordas nítidas. De acordo com a métrica VQM, o borramento linear produzido pelo filtro Gaussiano obteve um desempenho relativamente mais elevado em comparação a outras técnicas como o filtro anisotrópico, com exceção do primeiro nível de ruído. O VBM3D e NLM obtiveram pontuações ligeiramente mais altas que o filtro Gaussiano, mas muito aquém da restauração obtida pelo filtro de mediana. Pode ser observado pela Figura 12 que quanto mais intenso o ruído impulsivo, maior a disparidade de eficiência entre o filtro de mediana e as outras técnicas.

Na Figura 13 é mostrado o efeito das filtragens em um vídeo deteriorado pelo ruído impulsivo com  $\sigma = 40$ . Como indicado pelos resultados quantitativos, apenas o filtro de mediana foi capaz de removê-lo efetivamente com pequeno efeito colateral da filtragem. Nas outras técnicas as maiores pontuações foram obtidas como resultado de um efeito intenso de borramento generalizado para maior aproximação em relação ao

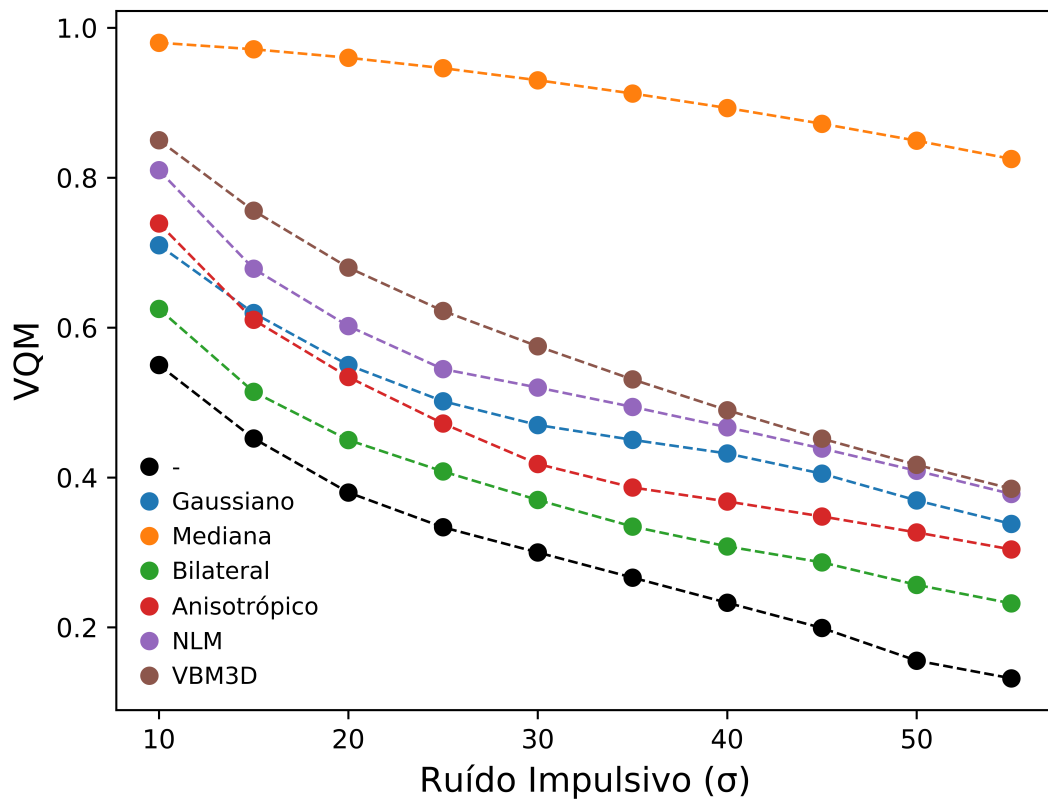


Figura 12 – Comparação dos algoritmos de filtragem para redução de ruído impulsivo.

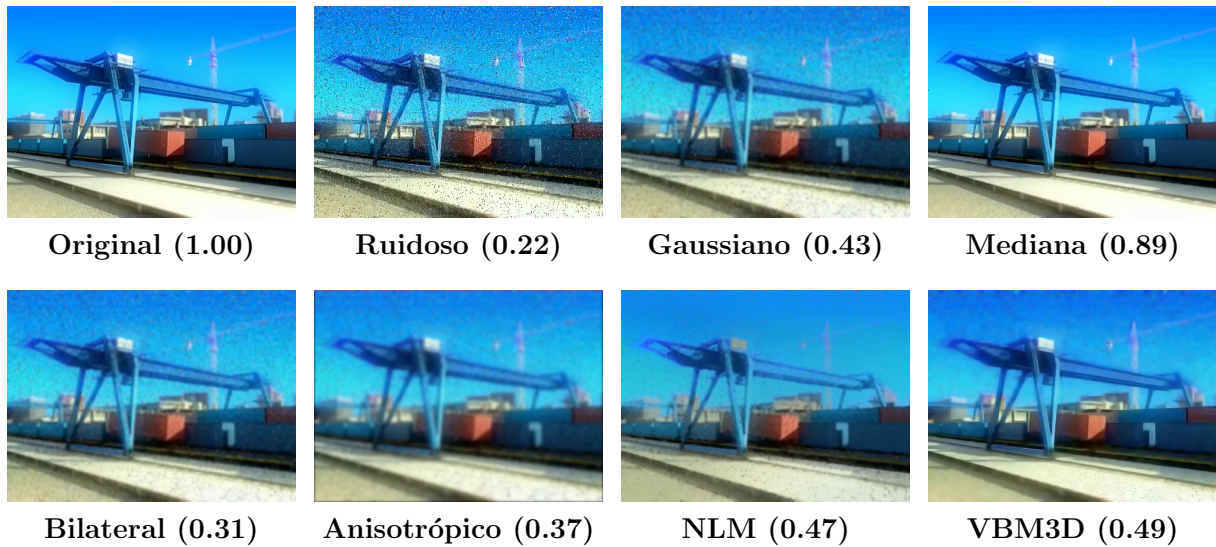


Figura 13 – Comparação visual da redução de ruído impulsivo com  $\sigma = 40$ .

vídeo de referência, mas o ruído ainda é claramente perceptível.

A Figura 14 apresenta os resultados em relação à remoção do ruído multiplicativo *speckle*. Assim como no ruído Gaussiano, pode-se constatar que o filtro NLM é o mais

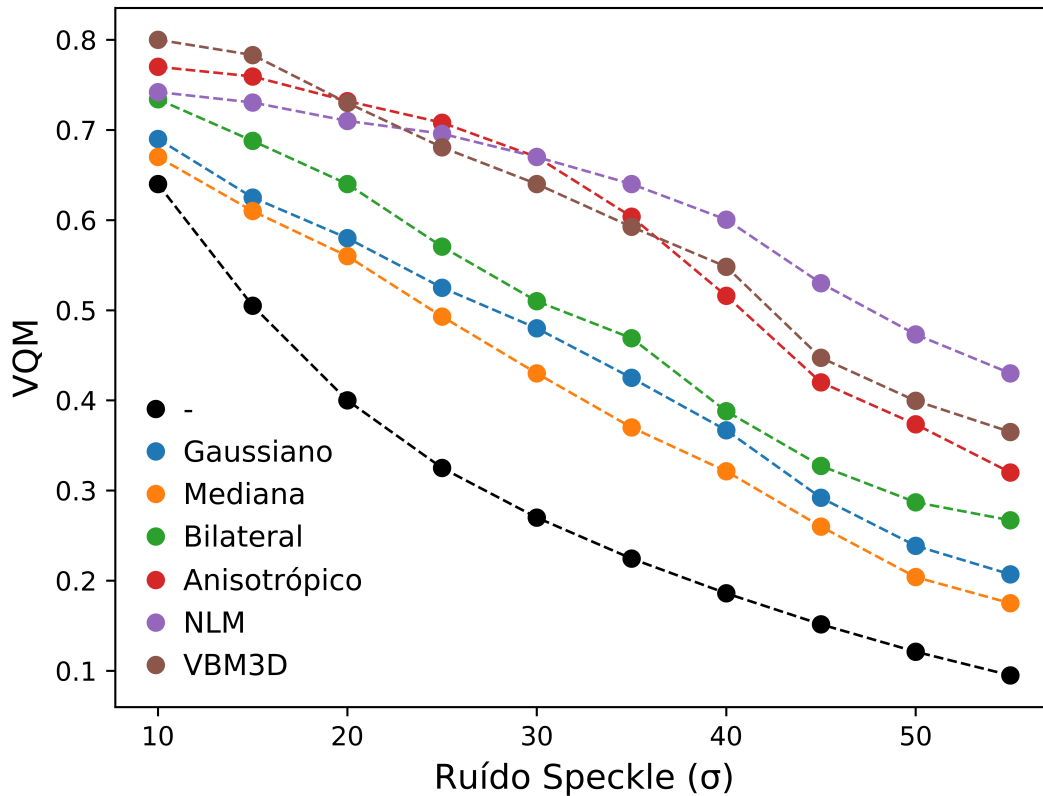


Figura 14 – Comparação dos algoritmos de filtragem para redução de ruído speckle.

eficaz nos maiores níveis de ruído, mas o seu melhor desempenho ocorre às custas da qualidade original, assim o VBM3D obtém melhores resultados com ruído abaixo de  $\sigma = 25$ , com pontuações médias razoavelmente inferiores em relação aos mesmos níveis de ruído Gaussiano. O filtro bilateral não obteve destaque para  $\sigma = 10$ , enquanto o filtro anisotrópico se mostrou bastante efetivo até  $\sigma = 30$ , tendo superado tanto o NLM como o VBM3D em  $\sigma = 25$ .

Podem ser observadas uma aceleração na queda do desempenho da filtragem nos maiores níveis de ruído após  $\sigma = 45$  para o filtro anisotrópico, NLM e VBM3D, um fenômeno que também ocorre nos resultados do ruído Gaussiano em menor escala. Tratando-se de um ruído multiplicativo, pode-se verificar uma maior dificuldade na restauração de regiões específicas do vídeo que se tornam saturadas com o ruído, como ilustrado na Figura 15, que contém os resultados para uma amostra de vídeo com ruído *speckle* de  $\sigma = 40$ . O VBM3D obteve um bom resultado visual, com exceção de várias falhas na região da água onde o ruído foi mantido. No caso do NLM, pode ser observado que praticamente todo o ruído é removido assim como no ruído Gaussiano, mas a melhor pontuação da métrica ocorre às custas de alguns detalhes do vídeo.

Resultados próximos da métrica podem indicar que em aplicações que não po-

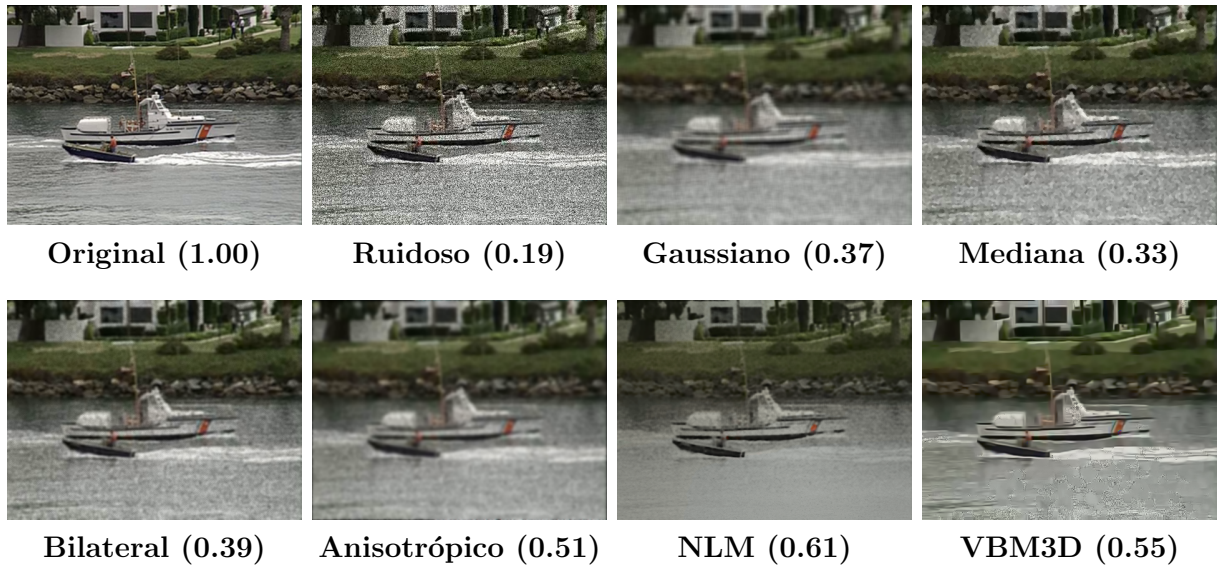


Figura 15 – Comparação visual da redução de ruído *speckle* com  $\sigma = 40$ .

dem depender de um tempo elevado de processamento para uma pequena diferença na qualidade resultante, o uso de filtros menos complexos pode ser mais vantajoso. O filtro Gaussiano e de mediana são capazes de processar os vídeos de maneira quase instantânea, permitindo sua aplicação em tempo real. O processamento pelos filtros bilateral e anisotrópico é um pouco mais prolongado, mas sem ultrapassar 1 minuto de duração para cada vídeo da base de dados, tornando-os satisfatórios nos cenários em que desempenham tão bem quanto técnicas mais complexas. Os filtros NLM e VBM3D demandam uma grande quantidade de tempo de processamento, podendo consumir vários minutos para um único vídeo, dessa forma, seu uso pode ser impraticável em certas aplicações.

A Tabela 3 apresenta o desvio padrão dos resultados obtidos pela métrica VQM associado às pontuações médias apresentadas na avaliação da filtragem, separado por cada técnica e tipo de ruído. Os valores indicam o nível de oscilação das pontuações de acordo com a variação do conteúdo original dos vídeos, que no pior caso de  $\sigma = 0.134$  para o filtro anisotrópico e ruído *speckle* a maior diferença para a média não é superior a 0.05 na pontuação pela métrica cujo intervalo é entre 0 e 1. Em relação ao tipo de ruído, os resultados para o ruído impulsivo tiveram a menor oscilação, seguido do ruído Gaussiano e o ruído *speckle* com as maiores diferenças da pontuação média. Pode ser observado que a pontuação mais estável é na avaliação da mesma instância de ruído em diferentes vídeos sem aplicação de filtragem. Em seguida, as técnicas lineares e não-lineares mais simples como o filtro Gaussiano, de mediana e bilateral apresentam menor discrepância entre as avaliações do vídeo processado em relação ao conteúdo original, enquanto os algoritmos não-locais que executam agrupamento de pixels como o NLM e VBM3D possuem maior variação para a filtragem do mesmo tipo e nível de ruído em diferentes vídeos. É possível constatar que a difusão anisotrópica é o filtro mais instável

Tabela 3 – Desvio padrão das pontuações VQM obtidas para cada ruído e técnica de filtragem.

<b>Ruído</b> <b>Filtro</b>	<b>Gaussiano</b>	<b>Impulsivo</b>	<b>Speckle</b>
—	0.053	0.012	0.067
Gaussiano	0.068	0.014	0.077
Mediana	0.060	0.005	0.072
Bilateral	0.079	0.017	0.091
Anisotrópico	0.101	0.026	0.134
NLM	0.091	0.022	0.115
VBM3D	0.084	0.019	0.101

devido ao realce das bordas, o que é altamente dependente do conteúdo e interfere na qualidade perceptível pela métrica. Por outro lado, houve poucas instâncias onde o filtro anisotrópico obteve o maior resultado.

A discrepância nas respostas de filtragem pela avaliação quantitativa indica que apesar de algumas distribuições serem mais próximas do ruído aditivo Gaussiano, aplicar um filtro de remoção de ruído de maneira generalizada geralmente não é eficaz, de forma que a adaptação do algoritmo de acordo com o modelo de ruído presente é desejável e até necessária dependendo da aplicação.

## 5.2 Detecção do tipo de ruído

A Tabela 4 apresenta a comparação de diferentes algoritmos de classificação para o modelo de detecção de ruído, de acordo com a metodologia descrita no capítulo 4. A princípio, foi observado um desempenho relativamente superior em relação às métricas de acurácia, precisão e pontuação F1 para o GBoost seguido do Random Forest com índices de 91% e 88% respectivamente. Com a abordagem de fusão dos atributos após a filtragem, todas as pontuações aumentaram com destaque para a SVM, MLP e GBoost com índices acima de 95%, onde a SVM obteve o melhor desempenho no problema multiclasse com uma acurácia geral de 96.45%, sendo assim o modelo de referência considerado em outros resultados relacionados de classificação neste capítulo.

A Tabela 5 apresenta a matriz de confusão das classes de ruído em relação à predição obtida nos conjuntos de teste da validação cruzada do modelo SVM. Cada valor representa a porcentagem de amostras que foram classificadas com determinada classe

Tabela 4 – Comparação dos algoritmos de classificação para aprendizado supervisionado com *10-fold Cross-Validation* na detecção de ruído.

Algoritmo	Sem filtragem			Filtragem e fusão de atributos		
	Acurácia	Precisão	F1	Acurácia	Precisão	F1
K-NN	84.40	84.46	84.23	92.13	92.40	92.07
C4.5	85.53	85.67	85.57	91.87	91.90	91.88
SVM	87.13	87.09	87.02	<b>96.45</b>	<b>96.47</b>	<b>96.45</b>
MLP	86.87	86.83	86.76	95.87	95.88	95.86
RF	88.67	88.67	88.66	93.53	93.57	93.55
GBoost	<b>91.07</b>	<b>91.08</b>	<b>91.07</b>	95.27	95.29	95.28

Tabela 5 – Matriz de confusão normalizada das classes de detecção de ruído com acurácia geral de 96.45% (modelo SVM).

Classe	Sem ruído	Impulsivo	Gaussiano	Speckle
Sem ruído	97.8%	0.2%	1.2%	0.8%
Ruído Impulsivo	0.8%	96.6%	1.8%	0.8%
Ruído Gaussiano	1.4%	0.8%	95.2%	2.6%
Ruído Speckle	0.8%	0.2%	2.8%	96.2%

indicada em cada coluna da tabela, em relação ao número de amostras com a classe verdadeira indicada nas linhas. Pode ser observado que a SVM treinou um modelo relativamente balanceado. Ainda assim, pelos verdadeiros positivos na diagonal principal da matriz é possível constatar que o ruído Gaussiano apresenta a menor taxa de acertos, com a maior quantidade de erros devido aos falsos positivos do ruído *speckle*. Por outro lado, a classe “sem ruído” obteve o maior número de verdadeiros positivos, dessa forma, na maior parte dos casos o detector é capaz de indicar a ausência de ruído, apesar de classificar 2% dos casos como ruído Gaussiano ou *speckle*. Outro ponto relevante é a elevada precisão ou baixa quantidade de falsos positivos para ruído impulsivo em comparação às outras classes, assim permitindo a remoção eficiente por um filtro de mediana na maioria das detecções, apesar de 3.4% de falsos negativos.

A Tabela 6 apresenta a avaliação da detecção em relação a cada classe distinta, considerando tanto o modelo aumentado pelo pré-processamento em que a SVM obteve os melhores resultados, e a abordagem sem fusão de atributos com o GBoost. Cada resultado da tabela inclui as amostras com todos os níveis de ruído considerados para

Tabela 6 – Comparação da detecção de ruído para cada classe.

Classe	Sem filtragem (GBoost)			Filtragem e fusão (SVM)		
	Precisão	TPR	F1	Precisão	TPR	F1
Sem ruído	95.77	95.00	95.38	97.02	97.80	97.41
Impulsivo	93.08	91.93	92.50	98.77	96.60	97.67
Gaussiano	86.53	87.14	86.83	94.26	95.20	94.73
Speckle	88.95	90.20	89.57	95.82	96.20	96.01

Tabela 7 – Comparação da detecção de ruído para cada nível (SVM).

Nível de ruído	Acurácia	Precisão	TPR	F1
$\sigma = 10$	78.67	80.11	78.67	78.76
$\sigma = 15$	94.50	94.80	94.50	94.58
$\sigma = 20$	97.20	97.45	97.20	97.24
$\sigma = 25$	97.67	97.88	97.67	97.71
$\sigma = 30$	97.71	97.95	97.71	97.76
$\sigma = 35$	98.00	98.21	98.00	98.05
$\sigma = 40$	98.44	98.59	98.44	98.48
$\sigma = 45$	98.40	98.52	98.40	98.43
$\sigma = 50$	98.55	98.70	98.55	98.58
$\sigma = 55$	98.67	98.77	98.67	98.69

a respectiva classe, considerando as métricas de precisão, taxa de verdadeiros positivos (TPR) e pontuação F1. Em relação às diferentes abordagens, foi verificada uma grande melhoria nos resultados para o modelo expandido, indicando a vantagem de empregar técnicas de filtragem na própria estratégia de detecção. A classe “sem ruído” e do ruído impulsivo apresentaram os melhores resultados, enquanto o ruído Gaussiano obteve as menores pontuações das métricas em relação às outras classes.

A Tabela 7 mostra os resultados das métricas de classificação considerando apenas os níveis de ruído para todas as amostras de vídeo e classes. Foi possível observar que nos menores níveis de ruído, especialmente no caso de  $\sigma = 10$ , a predição do modelo é reduzida significativamente, sendo responsável por diminuir as pontuações médias da classificação. A capacidade de predição aumenta ligeiramente quanto maior o nível de ruído, com pico em  $\sigma = 55$ .

A Figura 16 apresenta um gráfico ilustrando a variação do F1 para cada classe

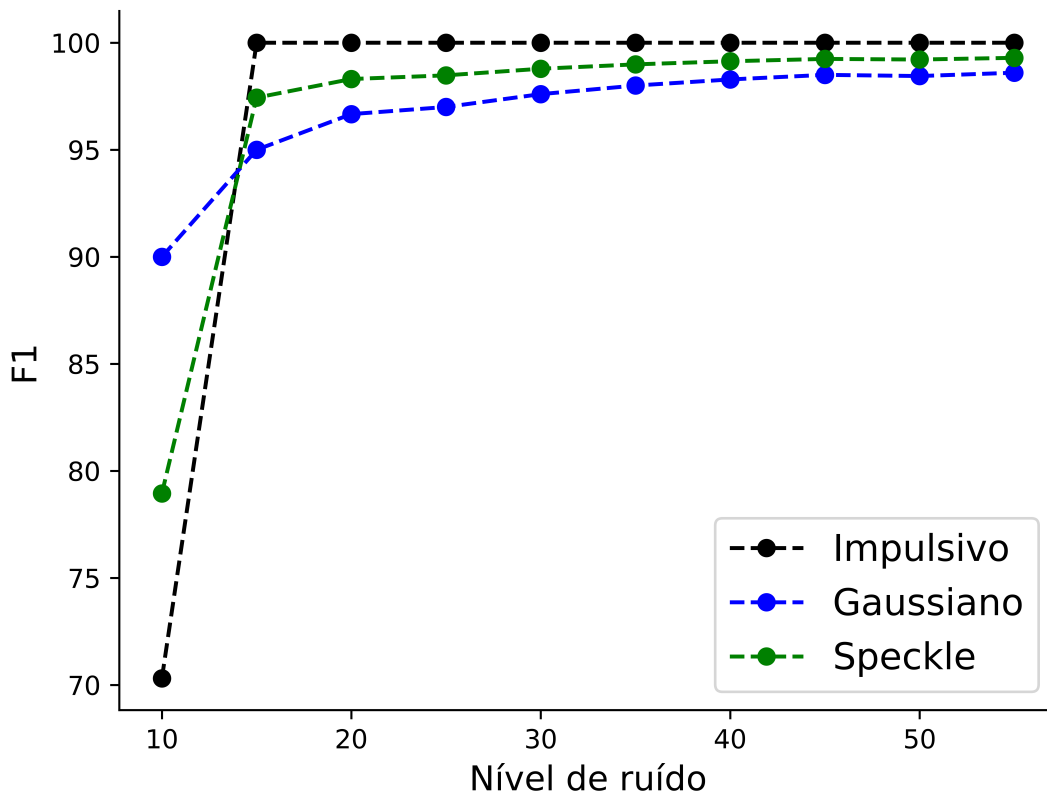


Figura 16 – Variação do F1 para cada classe em relação ao nível de ruído.

conforme os níveis de ruído, mostrando claramente a diferença da capacidade de predição no primeiro nível de ruído em relação ao restante. Pode-se observar que os ruídos impulsivo e *speckle* apresentam as menores pontuações em  $\sigma = 10$ , enquanto a classe de ruído Gaussiano possui o índice de 90% como o menor F1. Porém, sua taxa é ultrapassada pelos outros tipos nos outros níveis de ruído, com a classe do ruído impulsivo obtendo as maiores pontuações.

Deve ser destacado que a classificação executada pelo modelo neste trabalho não depende da presença de regiões homogêneas nas amostras como ocorre no trabalho em [49], que realiza a detecção dos mesmos tipos de ruído utilizando redes neurais. Em [49], a acurácia de 97% na classificação do ruído impulsivo apenas em amostras com mais áreas homogêneas se compara aos índices obtidos neste trabalho pela SVM, enquanto para o ruído Gaussiano e *speckle* a taxa foi menor que 89%, já neste trabalho foi 95% e 96% respectivamente. Outro ponto é que em [49] a acurácia foi inversamente proporcional ao nível de ruído, ou seja, o desempenho da classificação foi reduzido conforme o aumento de ruído nas diferentes classes, enquanto neste trabalho uma maior quantidade de ruído é o cenário ideal para favorecer o desempenho do detector.

Outros trabalhos relacionados no capítulo 3 obtêm índices de predição de mais de

Tabela 8 – Comparação dos algoritmos de regressão para aprendizado supervisionado com *10-fold Cross-Validation* na estimação de ruído.

Alg.	Sem filtragem				Filtragem e fusão			
	$\mu \sigma_e - \sigma_t $	$\sigma \sigma_e - \sigma_t $	MSE	PCC	$\mu \sigma_e - \sigma_t $	$\sigma \sigma_e - \sigma_t $	MSE	PCC
K-NN	3.74	3.70	27.73	0.9304	3.55	3.43	24.39	0.9391
C4.5	3.82	4.72	36.82	0.9102	3.58	4.41	32.32	0.9219
SVM	3.93	3.78	29.71	0.9262	3.91	3.61	28.32	0.9298
MLP	3.50	<b>3.11</b>	21.95	0.9453	3.32	<b>3.19</b>	21.20	0.9473
RF	3.27	3.41	22.30	0.9445	3.18	3.30	21.02	0.9477
GBoost	<b>3.25</b>	3.28	<b>21.34</b>	<b>0.9469</b>	<b>3.11</b>	3.25	<b>20.22</b>	<b>0.9498</b>

90%, mas seus resultados são pouco comparáveis a este trabalho devido a uma combinação de diferentes fatores: menos classes de ruído consideradas, requerimento de regiões homogêneas, classificação de imagens no lugar de vídeos, poucas amostras na base de dados e a não aplicação da validação cruzada na avaliação do classificador. Ao contrário da estimação de ruído, os trabalhos encontrados na área de detecção não permitem a reprodução de resultados ou dos métodos para experimentação em diferentes bases de dados.

### 5.3 Estimação do nível de ruído

Os algoritmos empregados na classificação do tipo de ruído foram utilizados para regressão pelo modelo, a fim de estimar o nível de ruído presente representado pelo desvio padrão ( $\sigma$ ) em cada amostra de vídeo degradado. Como descrito no capítulo 4, a estimação do ruído é realizada considerando apenas os vídeos com as classes de ruído impulsivo, Gaussiano e *speckle*, visto que a detecção é executada antes da estimação e nenhuma filtragem deve ser aplicada no caso da classe “sem ruído”. A Tabela 8 mostra os resultados obtidos em relação à diferença absoluta entre o nível estimado de ruído ( $\sigma_e$ ) e o valor verdadeiro ( $\sigma_t$ ), considerando a média ( $\mu$ ) e o desvio padrão das diferenças, o erro quadrático médio (MSE) e o coeficiente de correlação de Pearson (PCC).

Desta vez, apenas o GBoost alcançou os melhores resultados nas duas estratégias de processamento em relação às métricas de erro e correlação da estimação, enquanto o RF e a rede neural MLP obtiveram resultados próximos. No caso do desvio padrão das diferenças, a MLP obteve os menores valores, indicando menor oscilação das estimativas do nível de ruído. Pode ser observado que a contribuição da estratégia de filtragem não foi tão expressiva para a regressão em comparação aos resultados de classificação. Ainda assim, o GBoost foi capaz de obter um erro relativamente baixo e elevar o coeficiente de correlação para quase 95%.

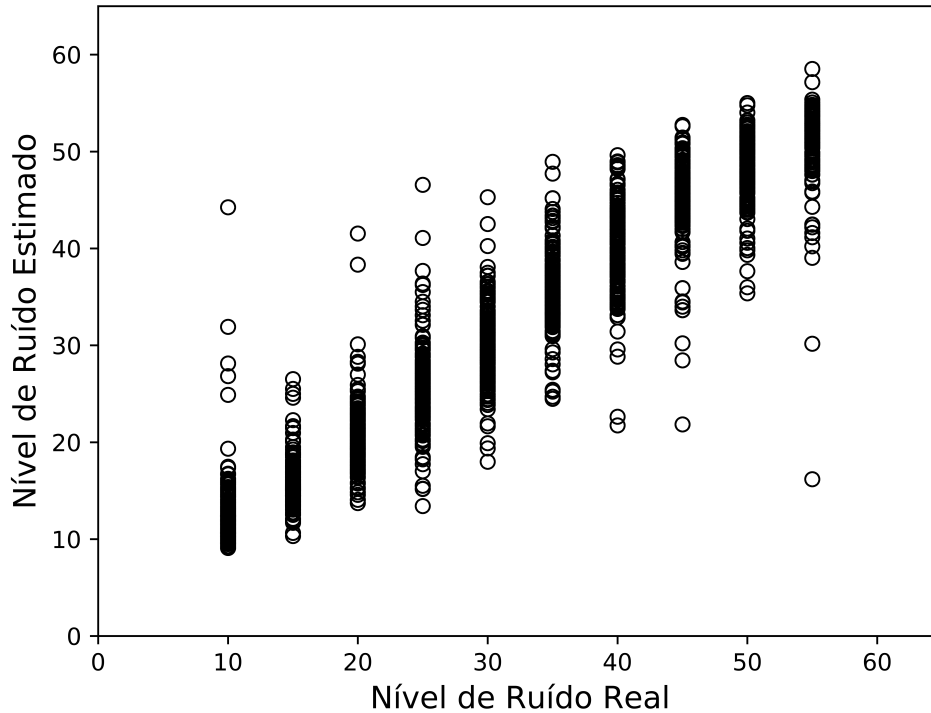


Figura 17 – Estimação do nível de ruído com coeficiente de correlação de Pearson de 95%.

A Figura 17 ilustra os níveis estimados de ruído pelo GBoost em relação aos respectivos níveis reais de ruído para cada amostra degradada. Pode ser observado que o primeiro nível de ruído possui o maior número de *outliers*, o que de certo modo se relaciona com os resultados de classificação apresentados na Tabela 7. Os casos de ruído com desvio padrão de  $\sigma = 55$  também possuem dois *outliers*, mas o desempenho médio da estimação é tão alto quanto os outros níveis.

A estimação do nível de ruído realizada pelo modelo do GBoost foi avaliada em relação à estimação por outros três métodos da literatura descritos no capítulo 3, que utilizam técnicas baseadas em análise de componentes principais (PCA) [55], transformada discreta de cosseno (DCT) [56] e transformada *wavelet* (WT) [54] respectivamente.

Na Tabela 9 é apresentada a comparação dos resultados obtidos separados para cada tipo de ruído, seguidos dos resultados médios da estimação considerando todas as amostras com ruído da base de dados. Pode ser observado que o modelo do GBoost obteve os melhores resultados pelas métricas em relação ao ruído impulsivo e *speckle*. Para o ruído Gaussiano o resultado foi mais baixo, o que também se relaciona com os resultados de classificação na Tabela 6. Por outro lado, o ruído Gaussiano foi a classe onde os métodos alternativos obtiveram os melhores resultados de estimação entre os três tipos de ruído. O método com WT obteve o menor resultado médio em todas das instâncias, sendo o único dos métodos testados com a suposição de que existe uma quantidade suficiente de regiões

Tabela 9 – Comparação do modelo de regressão do GBoost em relação a PCA, DCT e WT na estimação do nível de ruído para diferentes classes.

Classe	Método	$\mu \sigma_e - \sigma_t $	$\sigma \sigma_e - \sigma_t $	MSE	PCC
Impulsivo	PCA	4.44	3.35	30.95	0.9315
	DCT	4.27	3.34	29.39	0.9359
	WT	6.69	4.89	68.62	0.8635
	GBoost	<b>2.91</b>	<b>2.90</b>	<b>16.86</b>	<b>0.9607</b>
Gaussiano	PCA	<b>2.94</b>	<b>2.16</b>	<b>13.32</b>	<b>0.9686</b>
	DCT	3.90	2.94	23.81	0.9458
	WT	5.54	3.43	42.43	0.9117
	GBoost	3.48	3.89	27.27	0.9325
Speckle	PCA	6.22	4.27	56.95	0.8833
	DCT	4.18	3.18	27.57	0.9383
	WT	5.86	4.11	51.19	0.8929
	GBoost	<b>2.93</b>	<b>2.82</b>	<b>16.54</b>	<b>0.9597</b>
Todas	PCA	4.53	3.63	33.74	0.9255
	DCT	4.11	<b>3.16</b>	26.92	0.9396
	WT	6.03	4.21	54.08	0.8877
	GBoost	<b>3.11</b>	3.25	<b>20.22</b>	<b>0.9498</b>

homogêneas para estimação do ruído.

O método com PCA obteve os melhores resultados de estimação para o ruído Gaussiano com quase 97% de correlação, sendo também o maior índice em uma classe individual. No caso do PCA, como foi visto no capítulo 3, não há a suposição de que o vídeo contém áreas homogêneas; a suposição é de que existem estruturas de textura redundantes que são correlacionadas entre si enquanto o ruído é independente, onde a variância do ruído é computada como o menor autovalor da matriz de covariância. Dessa forma, pode ser observado que o PCA também obteve um resultado satisfatório na estimação do ruído impulsivo, que é um ruído aditivo como o Gaussiano. Para o ruído multiplicativo *speckle* o PCA obteve as menores pontuações entre os métodos incluindo a WT, visto que sua suposição é invalidada uma vez que o ruído é correlacionado com o conteúdo. O método DCT foi o mais equilibrado, tendo obtido o segundo melhor resultado em cada classe de ruído.

Dessa forma, os resultados se relacionam com as suposições de cada abordagem, enquanto o modelo do GBoost é o único método avaliado que utiliza aprendizado de máquina e possui potencial de ajustar a estimação quando há estruturas do conteúdo nas altas frequências ou quando o ruído é correlacionado, tendo assim obtido as melhores pontuações pelas métricas ao considerar todas as classes, com exceção do desvio padrão

Tabela 10 – Comparação da redução de ruído com VBM3D utilizando os níveis estimados como parâmetro de filtragem.

Nível de ruído		$\sigma_t = 10$		$\sigma_t = 30$		$\sigma_t = 50$	
Tipo	Estim.	$\bar{\sigma}_e$	VQM	$\bar{\sigma}_e$	VQM	$\bar{\sigma}_e$	VQM
Gauss.	—	—	0.706201	—	0.275950	—	0.158678
	$\sigma_p = 10$	10.00	0.824904	10.00	0.544777	10.00	0.174731
	$\sigma_p = 20$	20.00	0.813928	20.00	0.726039	20.00	0.204409
	$\sigma_p = 30$	30.00	0.809697	30.00	0.728219	30.00	0.399271
	$\sigma_p = 40$	40.00	0.796027	40.00	0.721528	40.00	0.474711
	$\sigma_p = 50$	50.00	0.762703	50.00	0.703950	50.00	0.481433
	$\sigma_p = 60$	60.00	0.720312	60.00	0.686193	60.00	0.474264
	PCA	10.37	<b>0.824512</b>	30.18	<b>0.728137</b>	49.51	0.481253
	DCT	10.62	0.824372	29.84	0.728120	50.04	<b>0.481441</b>
	WT	9.16	0.821695	29.07	0.726387	46.42	0.479328
GBoost	13.23	0.821372	30.92	0.727914	47.54	0.480191	
Speckle	—	—	0.639946	—	0.275782	—	0.124786
	$\sigma_p = 10$	10.00	0.799897	10.00	0.565112	10.00	0.211826
	$\sigma_p = 20$	20.00	0.776916	20.00	0.627342	20.00	0.256470
	$\sigma_p = 30$	30.00	0.721541	30.00	0.644582	30.00	0.347695
	$\sigma_p = 40$	40.00	0.668274	40.00	0.637588	40.00	0.383337
	$\sigma_p = 50$	50.00	0.631413	50.00	0.590671	50.00	0.403769
	$\sigma_p = 60$	60.00	0.597522	60.00	0.546101	60.00	0.374107
	PCA	11.98	0.798121	31.76	0.643775	52.61	0.402179
	DCT	8.84	0.796259	30.92	0.644570	50.58	<b>0.403864</b>
	WT	8.22	0.792540	28.62	0.643841	46.27	0.400682
GBoost	11.49	<b>0.798187</b>	30.56	<b>0.644650</b>	49.43	0.403576	

dos erros de estimação onde a DCT obteve um valor menor, apesar da média das diferenças ser mais alta que a do GBoost.

Para testar a validade dos resultados obtidos na estimação de ruído, foi utilizado o filtro VBM3D para executar a redução de ruído aplicando como parâmetro de entrada do desvio padrão esperado o valor do nível estimado de ruído, considerando as amostras de vídeo degradadas com ruído Gaussiano e *speckle*, visto na seção 5.1 que ele não é efetivo para o ruído impulsivo. Com os respectivos vídeos de referência, o desempenho da estimação pode ser comparado através uma métrica de referência completa como a VQM. A Tabela 10 apresenta as pontuações médias obtidas pela métrica VQM considerando múltiplos níveis de ruído como parâmetro de entrada ( $\sigma_p$ ) para a técnica VBM3D em três níveis de ruído nos vídeos com desvio padrão ( $\sigma_t$ ) de 10, 30 e 50. Os parâmetros fixados

de  $\sigma_p = 10$  a  $\sigma_p = 60$  são utilizados como linha de base. O valor médio estimado por PCA, DCT, WT e GBoost é mostrado em cada caso na coluna indicada por  $\bar{\sigma}_e$ .

É possível observar na Tabela 10 que o ajuste de filtragem do VBM3D com estimativas mais próximas do nível de referência de fato resulta em pontuações mais altas em relação às outras linhas de base, mas os resultados demonstram que uma estimativa diferente do nível real de ruído pode fornecer um resultado quantitativo ligeiramente superior na remoção de ruído pela métrica, como no caso de  $\sigma_t = 50$  e em  $\sigma_t = 30$  para o ruído *speckle*, onde o valor médio estimado é um pouco mais alto que o nível real de ruído, ou seja, com a aplicação da filtragem com uma intensidade maior do que a projetada no algoritmo VBM3D para o respectivo nível de ruído. Deve-se destacar que a estimativa média ( $\bar{\sigma}_e$ ) de um método pode ser mais próxima do nível real de ruído em comparação a outro método que possui um erro menor, visto que o erro considera as diferenças absolutas para o valor verdadeiro, enquanto isso não é indicado no valor médio por nível de ruído.

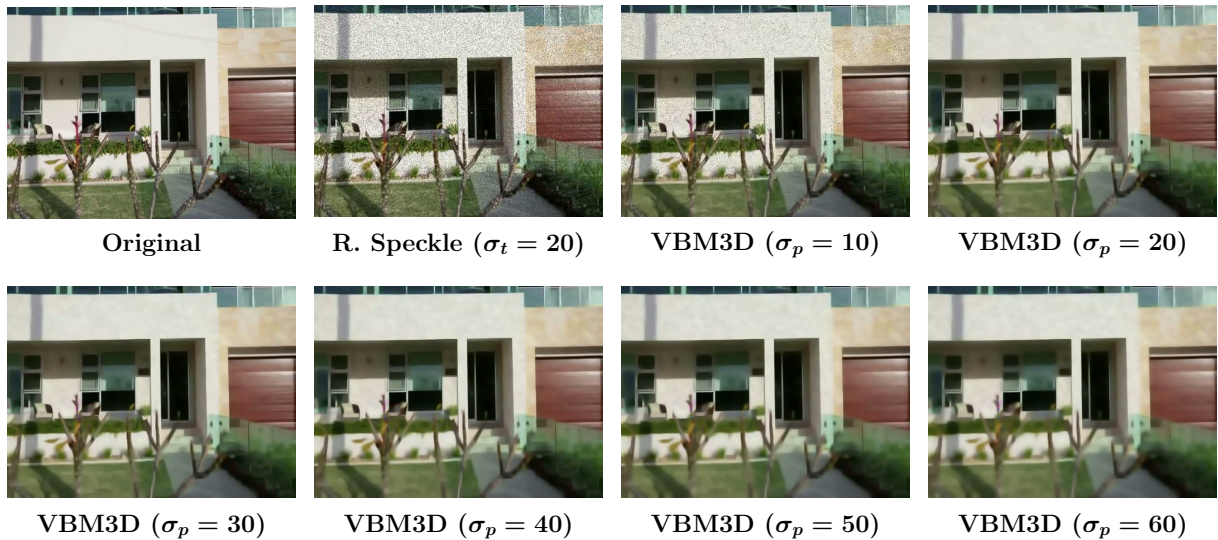


Figura 18 – Vídeo com ruído *speckle* ( $\sigma = 20$ ) processado pelo VBM3D com diferentes níveis como parâmetro de entrada.

Em relação aos diferentes métodos de estimação, na maioria dos casos o melhor estimador obteve uma pontuação média mais alta pela métrica VQM, apesar de uma diferença bastante reduzida em comparação aos resultados obtidos pelos outros parâmetros próximos de entrada. O método PCA com a melhor estimativa para o ruído Gaussiano e o modelo GBoost para o ruído multiplicativo obtiveram as maiores VQM associadas com exceção de  $\sigma_t = 50$ , onde a DCT forneceu uma redução de ruído quantitativamente superior com ajuste sobrestimado da filtragem. Mesmo a estimativa com maiores índices de erro realizada pela WT alcançou uma restauração superior às linhas de base adjacentes, e não muito discrepante do melhor caso. Nos casos com maior nível de ruído a diferença entre as pontuações aumenta para diferentes níveis de entrada, mas ao mesmo tempo o potencial de restauração da qualidade é limitado, o que é evidenciado pelas pontuações

mais baixas da métrica VQM.

Na comparação qualitativa a diferença entre parâmetros de entrada muito próximos não é perceptível, como pode ser constatado na Figura 18, que mostra respectivamente um quadro do vídeo original, da versão degradada pelo ruído *speckle* com  $\sigma_t = 20$ , e de amostras processadas utilizando diferentes níveis de ruído de  $\sigma_p = 10$  a  $\sigma_p = 60$  como parâmetros de entrada para o VBM3D. Pode ser observado que com  $\sigma_p = 10$  o ruído ainda é visível nas regiões homogêneas do quadro. Para  $\sigma_p = 20$ , o valor “verdadeiro” ou de referência do desvio padrão do ruído, praticamente todo o ruído é removido como esperado. A amostra processada com  $\sigma_p = 30$  é aparentemente igual a anterior, enquanto nas amostras seguintes pode ser verificada uma degradação dos detalhes originais devido ao maior efeito de borramento pela filtragem.

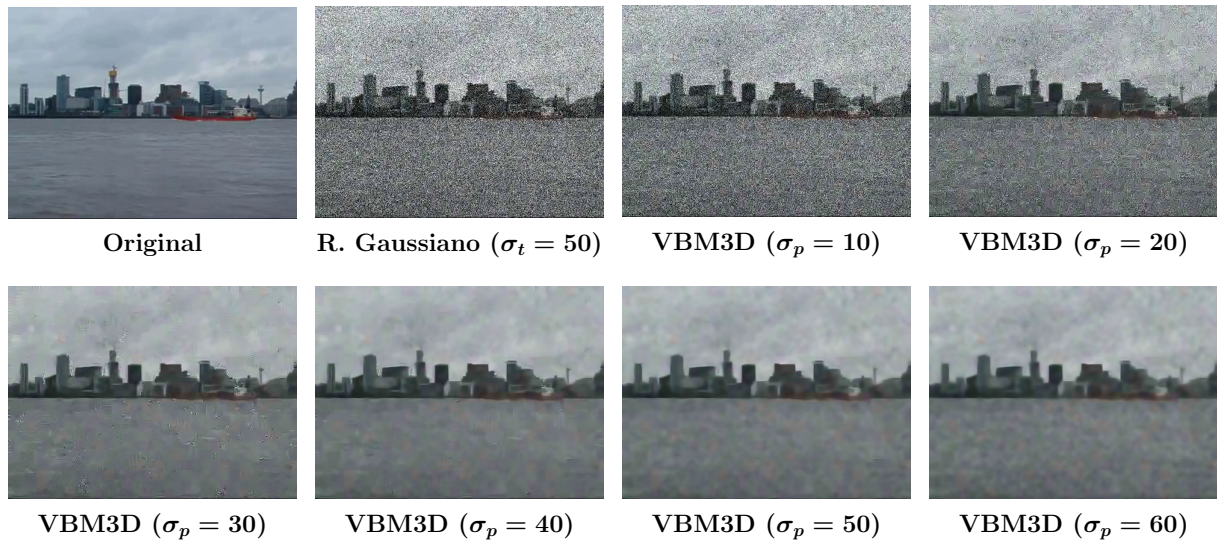


Figura 19 – Vídeo com ruído Gaussiano ( $\sigma = 50$ ) processado pelo VBM3D com diferentes níveis como parâmetro de entrada.

Um segundo exemplo é mostrado na Figura 19, onde o vídeo é degradado pelo ruído Gaussiano com  $\sigma_t = 50$ . O ruído é bastante visível até  $\sigma_p = 20$ , com algumas falhas locais em  $\sigma_p = 30$ , enquanto as amostras processadas de  $\sigma_p = 40$  a  $\sigma_p = 60$  são visualmente equivalentes. Quantitativamente, a diferença é claramente detectada por uma métrica de referência completa, como mostrado na Tabela 10. Dessa forma, não é viável comparar diferentes resultados de estimação de maneira qualitativa, sendo necessário recorrer à precisão das métricas para comparar e verificar a vantagem dos métodos de estimação. Por outro lado, as Figuras 18 e 19 indicam que para prejudicar visualmente o desempenho da restauração, o erro da estimação deve ser relativamente elevado.

Tabela 11 – Comparação do número de amostras para cada técnica de filtragem de referência de acordo com o tipo de ruído.

<b>Ruído</b> <b>Filtro</b>	<b>Sem ruído</b>	<b>Impulsivo</b>	<b>Gaussiano</b>	<b>Speckle</b>	<b>Total</b>
<b>Nenhum</b>	500	0	0	0	500
<b>Gaussiano</b>	0	0	0	0	0
<b>Mediana</b>	0	500	0	0	500
<b>Bilateral</b>	0	0	0	0	0
<b>Anisotrópico</b>	0	0	0	150	150
<b>NLM</b>	0	0	250	250	500
<b>VBM3D</b>	0	0	250	100	350

#### 5.4 Redução de ruído com seleção da técnica de filtragem

A Tabela 11 apresenta a quantidade de amostras associadas a cada técnica de filtragem de ruído de acordo com a maior pontuação pela métrica VQM obtida para cada tipo e nível de ruído nos experimentos de avaliação na seção 5.1, contendo 1500 vídeos divididos nas classes de ruído impulsivo, Gaussiano e *speckle* da avaliação e incluindo os 500 vídeos sem degradação de ruído que foram utilizados para detecção de ruído na seção 5.2, totalizando 2000 amostras.

No caso das amostras sem ruído, foi atribuído diretamente como filtro de referência a denominação “nenhum”, uma vez que sem qualquer processamento elas são equivalentes ao próprio vídeo de referência e a pontuação pela métrica VQM obtida é 1, o maior valor possível. No caso do ruído impulsivo, pode ser observado na Tabela 11 que todos os casos foram relacionados ao filtro de mediana, visto que ele consistentemente obteve as maiores pontuações em todos os níveis de ruído em comparação às outras técnicas. Para o ruído Gaussiano e *speckle* a rotulação foi feita entre as técnicas NLM e VBM3D, incluindo o filtro anisotrópico para o ruído multiplicativo de acordo com as pontuações médias observadas em cada nível de ruído. O filtro Gaussiano e bilateral não obtiveram maior pontuação pela métrica VQM em nenhum caso, dessa forma foram excluídos do processo de seleção. Pode-se observar que a base para seleção das técnicas de filtragem não é balanceada, com um número mais baixo de amostras para o filtro anisotrópico e VBM3D.

A Tabela 12 apresenta os resultados de seleção da técnica de filtragem utilizando as métricas de classificação, de acordo com a detecção e estimação de ruído realizadas em relação à técnica de referência atribuída ao respectivo tipo e nível verdadeiro de ruído. Como indicador, a primeira linha na tabela mostra o resultado da seleção utilizando a classe e nível de ruído reais ou de referência, com índices de 100% para as técnicas seleti-

Tabela 12 – Comparação da seleção da técnica de filtragem com diferentes métodos de detecção e estimação de ruído.

Detecção	Estimação	Acurácia	Precisão	F1
Referência ( $C_t$ )	Ref. ( $\sigma_t$ )	100.00	100.00	100.00
SVM	Ref. ( $\sigma_t$ )	97.35	97.38	97.36
Referência ( $C_t$ )	PCA	92.15	92.14	92.14
	DCT	93.25	93.28	93.27
	WT	91.80	91.78	91.79
	GBoost	<b>95.90</b>	<b>95.89</b>	<b>95.89</b>
SVM	PCA	89.65	89.70	89.67
	DCT	90.95	91.05	90.99
	WT	89.35	89.35	89.34
	GBoost	<b>93.25</b>	<b>93.27</b>	<b>93.26</b>

onadas, seguido do resultado para a detecção de ruído pelo modelo de classificação SVM da seção 5.2 utilizando também o nível real de ruído, com uma seleção correta da técnica de redução de ruído em mais de 97% dos casos. Na sequência, é comparada a seleção utilizando a estimação de ruído pelo modelo GBoost e os outros estimadores avaliados na seção 5.3, em combinação com a detecção de referência e pela SVM respectivamente, com os melhores índices resultantes na seleção com estimação pelo GBoost em ambos os casos. Deve-se ressaltar que os parâmetros de referência são para fins de comparação, enquanto o resultado a ser destacado é a seleção da técnica utilizando unicamente a detecção e estimação sem nenhuma informação prévia a respeito do vídeo alvo, evidenciando a seleção correta da técnica de redução de ruído mais adequada em 93.25% das amostras no caso do modelo SVM e GBoost.

As Tabelas 13, 14 e 15 representam matrizes de confusão da seleção das técnicas de ruído considerando respectivamente como entrada: a classe de ruído detectada pela SVM e o nível real de ruído; a classe real de ruído e o nível de ruído estimado pelo GBoost; e a classe e nível estimados por ambos os modelos. Cada matriz de confusão permite visualizar com maior detalhamento os resultados de predição indicados na Tabela 12 com acurácia de 97.35%, 95.90% e 93.25% respectivamente.

Pode ser observado que os verdadeiros positivos na seleção do filtro “nenhum” e de mediana nas Tabelas 13 e 15 com detecção de ruído equivalem aos verdadeiros positivos na matriz de confusão da Tabela 5 na seção 5.2 em relação à classificação dos tipos de ruído, visto que a falha na detecção da ausência de ruído ou do ruído impulsivo implica diretamente no erro da filtragem escolhida. No caso da Tabela 14 com a detecção

Tabela 13 – Matriz de confusão da seleção da filtragem pela detecção de ruído (SVM) e o nível real do ruído.

<b>Filtro</b>	<b>Nenhum</b>	<b>Mediana</b>	<b>Anisotrópico</b>	<b>NLM</b>	<b>VBM3D</b>
<b>Nenhum</b>	97.80%	0.20%	0.00%	0.00%	2.00%
<b>Mediana</b>	0.80%	96.60%	0.00%	0.00%	2.60%
<b>Anisotrópico</b>	0.00%	0.67%	96.67%	0.00%	2.67%
<b>NLM</b>	0.00%	0.20%	0.00%	99.80%	0.00%
<b>VBM3D</b>	3.14%	0.86%	1.43%	0.00%	94.57%

Tabela 14 – Matriz de confusão da seleção da filtragem pelo tipo real do ruído e o nível estimado de ruído (GBoost).

<b>Filtro</b>	<b>Nenhum</b>	<b>Mediana</b>	<b>Anisotrópico</b>	<b>NLM</b>	<b>VBM3D</b>
<b>Nenhum</b>	100.00%	0.00%	0.00%	0.00%	0.00%
<b>Mediana</b>	0.00%	100.00%	0.00%	0.00%	0.00%
<b>Anisotrópico</b>	0.00%	0.00%	83.33%	8.00%	8.67%
<b>NLM</b>	0.00%	0.00%	3.40%	94.40%	2.20%
<b>VBM3D</b>	0.00%	0.00%	1.71%	6.57%	91.71%

Tabela 15 – Matriz de confusão da seleção da filtragem pela detecção de ruído (SVM) e o nível estimado de ruído (GBoost).

<b>Filtro</b>	<b>Nenhum</b>	<b>Mediana</b>	<b>Anisotrópico</b>	<b>NLM</b>	<b>VBM3D</b>
<b>Nenhum</b>	97.80%	0.20%	0.00%	0.00%	2.00%
<b>Mediana</b>	0.80%	96.60%	0.00%	0.00%	2.60%
<b>Anisotrópico</b>	0.00%	0.67%	80.00%	8.00%	11.33%
<b>NLM</b>	0.00%	0.20%	3.40%	94.20%	2.20%
<b>VBM3D</b>	3.14%	0.86%	3.14%	6.57%	86.29%

de referência, a seleção de nenhuma técnica ou do filtro de mediana é sempre correta, não sendo prejudicada pelo erro na estimação visto que a técnica de referência para esses casos é a mesma em todos os níveis de ruído. Já para as outras técnicas, pode-se constatar que tanto a detecção como a estimação interferem na técnica selecionada. A taxa de verdadeiros positivos do filtro anisotrópico é reduzida significativamente de 96% para 83% somente em consequência do erro no nível de ruído estimado, tendo em vista o menor número de amostras rotuladas e o intervalo restrito de níveis de ruído associados à sua maior pontuação no ruído *speckle*. Assim, os falsos negativos do filtro anisotrópico

Tabela 16 – Comparação do parâmetro com o valor de referência na seleção correta da técnica de filtragem de ruído.

Parâmetro	$\mu p_{sel} - p_{ref} $	$\sigma p_{sel} - p_{ref} $	MSE	PCC
Mediana ( <i>niters</i> )	0.1733	0.2245	0.0804	0.9556
Anisotrópico ( <i>alpha</i> )	0.0360	0.0568	0.0045	0.8543
NLM ( <i>h</i> )	2.95	3.57	21.42	0.9033
VBM3D ( $\sigma$ )	2.64	3.86	21.86	0.9507

ocorrem em decorrência da seleção do filtro NLM e em maior parte do VBM3D para níveis mais baixos de ruído, onde a estimação é menos eficiente.

No pior caso da Tabela 15, a seleção do NLM apresenta uma taxa de verdadeiros positivos acima de 94%, tendo em vista que a técnica possui a maior adequação de maneira contínua nos níveis mais altos de ruído. A seleção do VBM3D apresenta mais erros que o NLM e menos que o filtro anisotrópico, com muitos falsos positivos para o NLM quando a estimação é realizada pelo GBoost. Apesar do erro da seleção em relação à técnica de referência para determinada instância, é necessária a análise dos efeitos posteriores na redução de ruído por uma técnica não-ótima para determinar se a perda na restauração é proporcional ao erro na predição.

A Tabela 16 apresenta a comparação do parâmetro associado à técnica de filtragem selecionada de acordo com a detecção e estimação de ruído realizada pelos modelos SVM e GBoost, considerando a média e o desvio padrão das diferenças absolutas, e o MSE e PCC entre o parâmetro de referência ( $p_{ref}$ ) obtido pela estimação correta do nível de ruído e o parâmetro selecionado ( $p_{sel}$ ). Os resultados consideram apenas as instâncias onde a técnica foi selecionada corretamente, pois não é possível comparar parâmetros de técnicas diferentes. Pode ser observado que no caso da técnica VBM3D, os resultados pelas métricas de regressão do parâmetro selecionado se assemelham aos da estimação do nível de ruído, uma vez que o parâmetro de entrada é o próprio desvio padrão esperado do ruído, enquanto o parâmetro  $h$  do NLM apresenta resultados próximos, com um erro de diferença relativamente maior apesar do MSE mais baixo. No caso do filtro de mediana e anisotrópico, os parâmetros possuem escala e características diferenciadas, alterando assim a proporção das pontuações. O parâmetro *alpha* do filtro anisotrópico oscila entre zero e um, além disso, foi observado que o menor número de amostras para a técnica em conjunto com o intervalo limitado de oscilação dos valores associados aos níveis de ruído torna o coeficiente de correlação não adequado para sua avaliação. Para o filtro de mediana, tendo empregado um parâmetro fixo de janela 3x3 pixels para menor borramento, o parâmetro utilizado relaciona o número de iterações aplicadas que é uma variável discreta, onde é selecionada 1 iteração na maior parte dos casos até os níveis mais intensos de ruído com

Tabela 17 – Comparação da redução de ruído pela métrica VQM com seleção da técnica de filtragem usando diferentes métodos de detecção e estimação de ruído.

Filtro \ Ruído					Total	
	Sem ruído	Impulsivo	Gaussiano	Speckle	(2000)	(1500)
Nenhum	1.0000	0.3002	0.3138	0.2918	0.4765	0.3020
Gaussiano	0.9032	0.4846	0.4595	0.4430	0.5726	0.4624
Mediana	0.9193	0.9139	0.4011	0.4093	0.6609	0.5748
Bilateral	0.9109	0.3785	0.4945	0.4881	0.5680	0.4537
Anisotrópico	0.8909	0.4507	0.5960	0.5872	0.6312	0.5447
NLM	0.9098	0.5342	0.6495	0.6222	0.6789	0.6020
VBM3D	0.9263	0.5757	0.6639	0.5986	0.6911	0.6128
$S(C_t, \sigma_t)$	1.0000	0.9139	0.6888	0.6367	0.8099	0.7465
$S(\text{SVM}, \sigma_t)$	0.9984	0.9071	0.6823	0.6343	0.8055	0.7412
$S(C_t, \text{PCA})$	<b>1.0000</b>	0.9104	<b>0.6826</b>	0.6209	0.8035	0.7380
$S(C_t, \text{DCT})$	<b>1.0000</b>	0.9105	0.6802	0.6265	0.8043	0.7391
$S(C_t, \text{WT})$	<b>1.0000</b>	0.9088	0.6772	0.6234	0.8023	0.7364
$S(C_t, \text{GB})$	<b>1.0000</b>	<b>0.9116</b>	0.6802	<b>0.6299</b>	<b>0.8054</b>	<b>0.7406</b>
$S(\text{SVM}, \text{PCA})$	<b>0.9984</b>	0.9031	<b>0.6760</b>	0.6188	0.7991	0.7326
$S(\text{SVM}, \text{DCT})$	0.9982	0.9037	0.6741	0.6243	0.8001	0.7340
$S(\text{SVM}, \text{WT})$	0.9962	0.9019	0.6710	0.6212	0.7976	0.7314
$S(\text{SVM}, \text{GB})$	<b>0.9984</b>	<b>0.9048</b>	0.6737	<b>0.6276</b>	<b>0.8011</b>	<b>0.7354</b>

3 iterações.

A Tabela 17 apresenta as pontuações médias da métrica VQM obtidas com a seleção das técnicas de filtragem em relação a cada classe de ruído. Na última coluna é mostrada a média dos resultados considerando o conjunto com todas as 2000 amostras, e o conjunto com apenas as 1500 amostras que contém ruído. Similarmente à Tabela 12, as linhas são divididas em três seções: a primeira onde o nível real de ruído é considerado; a segunda onde a classe verdadeira é considerada, com diferentes métodos de estimação; e na terceira onde se utiliza a detecção do modelo SVM, novamente comparando os estimadores distintos. A abordagem de seleção é denotada pelo operador  $S$  onde o primeiro parâmetro se refere à classe de ruído ( $C_t$  ou detecção da SVM) e o segundo ao nível de ruído ( $\sigma_t$  ou estimação por PCA, DCT, WT ou GBoost). Na Figura 20 as pontuações gerais do conjunto de amostras com ruído são ilustradas em relação às técnicas individuais, à seleção de referência e à seleção pela detecção e estimação deste trabalho, onde os resultados pela

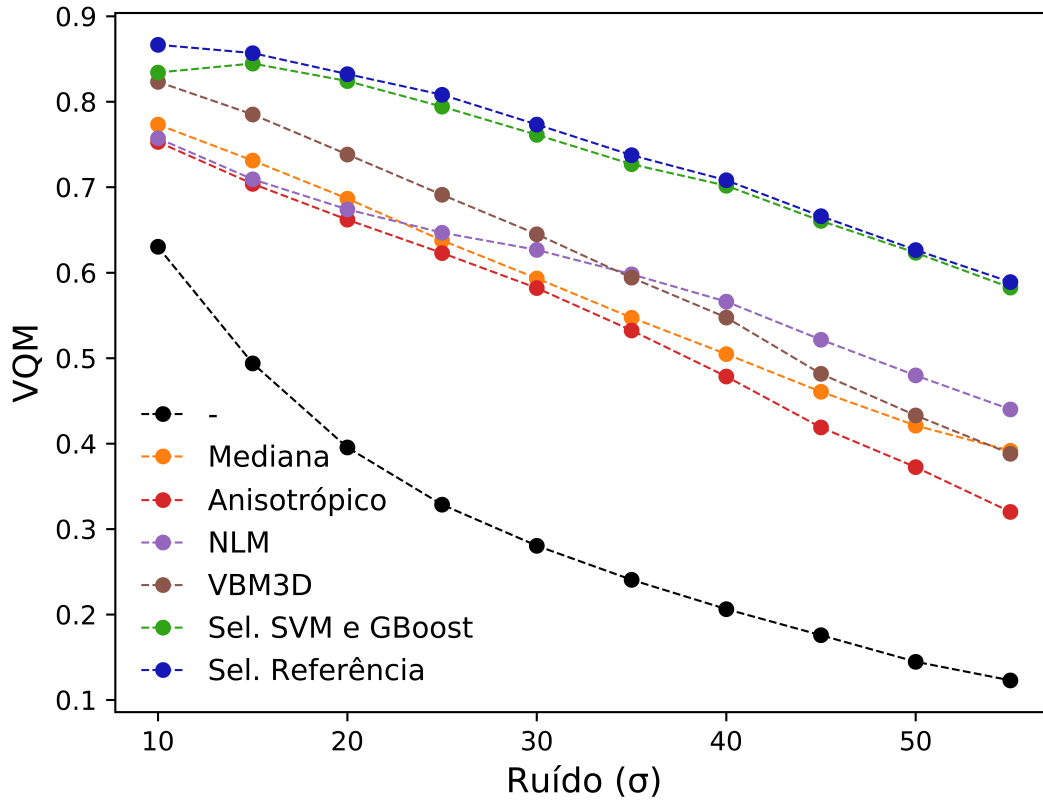


Figura 20 – Comparação da redução de ruído pela métrica VQM com seleção da técnica de filtragem utilizando detecção (SVM) e estimação (GBoost) de ruído.

métrica VQM são separados entre os diferentes níveis de ruído.

Como linhas de base, a Tabela 17 contém as pontuações obtidas com a seleção direta de uma única técnica de remoção de ruído com ajuste pelo nível real, ou seja, uma sumarização dos resultados da seção 5.1. A diferença é a coluna de pontuações para a classe “sem ruído”, onde as técnicas foram aplicadas nos vídeos sem presença de ruído considerando o menor nível de  $\sigma = 10$ , obtendo assim as pontuações das métricas em relação à “degradação da filtragem” com um intervalo de pontuações entre 0.9 e 0.93. Para o filtro “nenhum”, a pontuação obtida nas amostras sem ruído é 1, enquanto para os vídeos ruidosos ela equivale à pontuação obtida sem aplicação de filtragem. Na seleção de uma técnica única, é possível observar na Tabela 17 as maiores pontuações globais para o VBM3D, seguido do NLM, filtro de mediana, anisotrópico, Gaussiano e bilateral, onde os dois últimos com os menores resultados não foram atribuídos a nenhuma instância na relação de técnicas de referência. Pode ser constatado que na média geral, o filtro de mediana obteve um resultado superior ao filtro anisotrópico devido à eficiência no caso do ruído impulsivo.

Nas seleções considerando a classe real de ruído é obtida a pontuação máxima para

a classe “sem ruído”, além de um aumento considerável nas pontuações com aproximação dos maiores resultados possíveis pela seleção de referência denotada por  $S(C_t, \sigma_t)$ , em comparação às seleções realizadas com a detecção de ruído pela SVM que é sujeita a erros de classificação. Com a variação do estimador do nível de ruído, pode-se constatar que as pontuações obtidas seguem uma proporção similar à observada nos resultados de estimação da seção 5.3. A estimação pelo método com PCA e DCT resultaram em uma redução de ruído Gaussiano ligeiramente mais eficaz pela métrica VQM, visto que para esse tipo de ruído elas também apresentam uma maior correlação com o nível real de ruído. Para o ruído impulsivo e *speckle*, a redução de ruído com estimação pelo GBoost obteve a pontuação mais alta, resultando também em um melhor resultado geral em comparação aos outros métodos de estimação.

Apesar dos erros de seleção da técnica observados na Tabela 12, as pontuações resultantes da restauração na Tabela 17 com a detecção e estimação de ruído possuem uma diferença menor que 0.01 em relação à seleção de referência, ao mesmo tempo em que a diferença para a filtragem sem seleção da técnica é substancial, como evidenciado no gráfico da Figura 20. Assim, os resultados indicam que enquanto os modelos podem estimar um tipo e nível de ruído distintos e selecionar uma técnica ou parâmetro de filtragem diferentes da referência, o erro não possui o mesmo peso que uma seleção “aleatória”, pois a pontuação pela seleção resultante é relativamente mais próxima da referência.

## 6 CONCLUSÕES E TRABALHOS FUTUROS

A aquisição de vídeos é suscetível a diferentes tipos de ruídos que resultam na degradação de sua qualidade. Diversos trabalhos na área de remoção de ruído consideram uma generalização para o modelo do ruído aditivo Gaussiano, apesar da predominância de outros tipos de ruído em aplicações reais como o ruído impulsivo e o multiplicativo. Considerando estes modelos distintos de ruído, o trabalho avaliou 6 técnicas de filtragem em 12 bases de dados de vídeos utilizando a métrica de qualidade VQM, uma métrica específica para vídeos que possui alta correlação com a avaliação humana. Os experimentos mostraram variações nas respostas de filtragem conforme o tipo e nível de ruído presente nos vídeos, indicando a necessidade de análise do ruído a fim de adaptar a técnica de remoção de acordo com as características do ruído.

A partir disso, foi desenvolvido um detector e um estimador com modelos de aprendizado supervisionado, visando determinar o tipo e a quantidade de ruído incluindo a sua ausência, sem nenhuma informação prévia do vídeo alvo. Foi empregada uma abordagem de fusão dos atributos extraídos após a filtragem do vídeo, uma estratégia que aumentou consideravelmente a acurácia geral da detecção para 96.45% com a máquina de vetor de suporte, enquanto o modelo com GBoost obteve a melhor regressão na estimação do nível de ruído com uma correlação de 94.98%. Nenhum dos modelos requer a existência de regiões homogêneas no vídeo e apresentam maior desempenho em níveis mais altos de ruído, em contraste a outras propostas da literatura.

De forma a validar o detector e estimador desenvolvidos, foi feita a seleção da técnica de redução de ruído com base nas maiores pontuações pela métrica VQM obtidas para cada tipo e nível de ruído, resultando na seleção correta da técnica de filtragem em 93.25% dos casos, enquanto a pontuação geral obtida com a redução de ruído selecionada foi bastante próxima da seleção de referência e consideravelmente maior que na restauração com apenas uma das técnicas mais eficientes como a NLM ou VBM3D. Na comparação da estimação pelo modelo do GBoost em relação a outros estimadores com PCA e DCT, a proposta obteve resultados comparáveis mas não superiores para o ruído Gaussiano, sendo mais eficaz nos tipos impulsivo e multiplicativo e obtendo melhores resultados gerais na seleção das técnicas e na filtragem resultante pela métrica VQM.

As possibilidades de trabalhos futuros incluem a consideração de novas técnicas de restauração e de outras distribuições de ruído para aprimorar a capacidade e generalização do sistema adaptativo, levando em conta o aumento da complexidade na classificação com um número elevado de classes.

## REFERÊNCIAS

- [1] BOVIK, A. C. *Handbook of image and video processing*. [S.l.]: Academic press, 2010.
- [2] GONZALEZ, R. C.; WOODS, R. E. *Digital image processing*. [S.l.]: Upper Saddle River, NJ: Prentice Hall, 2012.
- [3] SHAO, L. et al. From heuristic optimization to dictionary learning: a review and comprehensive comparison of image denoising algorithms. *IEEE Transactions on Cybernetics*, IEEE, v. 44, n. 7, p. 1001–1013, 2014.
- [4] AMER, A.; DUBOIS, E. Fast and reliable structure-oriented video noise estimation. *IEEE Transactions on Circuits and Systems for Video Technology*, IEEE, v. 15, n. 1, p. 113–118, 2005.
- [5] JI, H. et al. Robust video denoising using low rank matrix completion. In: CITESEER. *CVPR*. [S.l.], 2010. p. 1791–1798.
- [6] MARQUES, O. *Digital video processing techniques and applications*. Wiley-IEEE Press, 2013.
- [7] VASEGHI, S. V. *Advanced digital signal processing and noise reduction*. [S.l.]: John Wiley & Sons, 2008.
- [8] FAROOQUE, M. A.; ROHANKAR, J. S. Survey on various noises and techniques for denoising the color image. *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, v. 2, n. 11, p. 217–221, 2013.
- [9] BOYAT, A. K.; JOSHI, B. K. A review paper: Noise models in digital image processing. *arXiv preprint arXiv:1505.03489*, 2015.
- [10] GONZALEZ, R. C.; WOODS, R. E. *Digital image processing*. Prentice Hall, 2008.
- [11] FILHO, M. O.; NETO, H. V. *Processamento digital de imagens*. Rio de Janeiro: Brasport, 1999.
- [12] JEON, G. Color image enhancement by histogram equalization in heterogeneous color space. *Int. J. Multimedia Ubiquitous Eng*, Citeseer, v. 9, n. 7, p. 309–318, 2014.
- [13] WANG, C. et al. Video enhancement using adaptive spatio-temporal connective filter and piecewise mapping. *EURASIP Journal on Advances in Signal Processing*, Springer International Publishing, v. 2008, n. 1, p. 1, 2008.
- [14] BUADES, A.; COLL, B.; MOREL, J.-M. A non-local algorithm for image denoising. In: IEEE. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. [S.l.], 2005. v. 2, p. 60–65.
- [15] DAI, J. et al. Color video denoising based on combined interframe and intercolor prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, IEEE, v. 23, n. 1, p. 128–141, 2013.

- [16] SZELISKI, R. *Computer vision: algorithms and applications*. [S.l.]: Springer Science & Business Media, 2010.
- [17] PHAM, T. Q.; VLIET, L. J. V. Separable bilateral filtering for fast video preprocessing. In: IEEE. *2005 IEEE International Conference on Multimedia and Expo*. [S.l.], 2005. p. 4–pp.
- [18] DURAND, F.; DORSEY, J. Fast bilateral filtering for the display of high-dynamic-range images. In: ACM. *ACM transactions on graphics (TOG)*. [S.l.], 2002. v. 21, n. 3, p. 257–266.
- [19] MAHMOUDI, M.; SAPIRO, G. Fast image and video denoising via nonlocal means of similar neighborhoods. *IEEE signal processing letters*, IEEE, v. 12, n. 12, p. 839–842, 2005.
- [20] REN, J. et al. Illumination-invariant non-local means based video denoising. In: IEEE. *2012 19th IEEE International Conference on Image Processing*. [S.l.], 2012. p. 1185–1188.
- [21] BUADES, A.; COLL, B.; MOREL, J.-M. Denoising image sequences does not require motion estimation. In: IEEE. *IEEE Conference on Advanced Video and Signal Based Surveillance, 2005*. [S.l.], 2005. p. 70–74.
- [22] LUO, E. et al. Adaptive non-local means for multiview image denoising: Searching for the right patches via a statistical approach. In: IEEE. *2013 IEEE International Conference on Image Processing*. [S.l.], 2013. p. 543–547.
- [23] DABOV, K. et al. Image denoising with block-matching and 3d filtering. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning*. [S.l.], 2006. v. 6064, p. 606414.
- [24] BROWN, R. G.; HWANG, P. Y. et al. *Introduction to random signals and applied Kalman filtering*. [S.l.]: Wiley New York, 1992. v. 3.
- [25] DABOV, K.; FOI, A.; EGIAZARIAN, K. Video denoising by sparse 3d transform-domain collaborative filtering. In: *Signal Processing Conference, 2007 15th European*. [S.l.: s.n.], 2007. p. 145–149.
- [26] FARIAS, M. C. *Video quality metrics*. [S.l.]: INTECH Open Access Publisher, 2010.
- [27] WANG, Y. Survey of objective video quality measurements. 2006.
- [28] WINKLER, S.; MOHANDAS, P. The evolution of video quality measurement: from psnr to hybrid metrics. *IEEE Transactions on Broadcasting*, IEEE, v. 54, n. 3, p. 660–668, 2008.
- [29] KOTEVSKI, Z. G.; MITREVSKI, P. J. Performance assessment of metrics for video quality estimation. 2010.
- [30] HUYNH-THU, Q.; GHANBARI, M. Scope of validity of psnr in image/video quality assessment. *Electronics letters*, IET, v. 44, n. 13, p. 800–801, 2008.

- [31] WANG, Z. et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, IEEE, v. 13, n. 4, p. 600–612, 2004.
- [32] WANG, Z.; LU, L.; BOVIK, A. C. Video quality assessment based on structural distortion measurement. *Signal processing: Image communication*, Elsevier, v. 19, n. 2, p. 121–132, 2004.
- [33] PINSON, M. H.; WOLF, S. A new standardized method for objectively measuring video quality. *IEEE Transactions on broadcasting*, IEEE, v. 50, n. 3, p. 312–322, 2004.
- [34] GARNETT, R. et al. A universal noise removal algorithm with an impulse detector. *IEEE Transactions on image processing*, IEEE, v. 14, n. 11, p. 1747–1754, 2005.
- [35] VATOLIN, D.; NOSKOV, A. *Brightness Flicking Metric*. 2002. Disponível em: <[http://www.compression.ru/video/quality\\_measure/metric\\_plugins/bfm\\_en.htm](http://www.compression.ru/video/quality_measure/metric_plugins/bfm_en.htm)>. Acesso em: 15.8.2018.
- [36] FARIAS, M. C.; MITRA, S. K. No-reference video quality metric based on artifact measurements. In: IEEE. *IEEE International Conference on Image Processing 2005*. [S.l.], 2005. v. 3, p. III–141.
- [37] AGARWAL, S. K.; KUMAR, P. Denoising of a mixed noise color image through special filter. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, v. 9, n. 1, p. 159–176, 2016.
- [38] RANTANEN, H. et al. Color video signal processing with median filters. *IEEE transactions on consumer electronics*, IEEE, v. 38, n. 3, p. 157–161, 1992.
- [39] GOKILAVANI, C. et al. Comparative results performance analysis of various filters used to remove noises in retinal images. In: IEEE. *Green Engineering and Technologies (IC-GET), 2015 Online International Conference on*. [S.l.], 2015. p. 1–5.
- [40] SUTOUR, C.; DELEDALLE, C.-A.; AUJOL, J.-F. Adaptive regularization of the nl-means: Application to image and video denoising. *IEEE Transactions on image processing*, IEEE, v. 23, n. 8, p. 3506–3521, 2014.
- [41] ARIAS, R. L. B.; FELINTO, A. S. Video denoising quality assessment for different noise distributions. In: IEEE. *2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*. [S.l.], 2018. p. 64–71.
- [42] PYATYKH, S.; HESSER, J.; ZHENG, L. Image noise level estimation by principal component analysis. *IEEE Transactions on Image Processing*, IEEE, v. 22, n. 2, p. 687–699, 2013.
- [43] GHOUSE, M.; SIDDAPPA, M. Adaptive techniques based high impulsive noise detection and reduction of a digital image. *Journal of Theoretical & Applied Information Technology*, v. 24, n. 1, 2011.
- [44] RAY, K. Unsupervised edge detection and noise detection from a single image. *Pattern Recognition*, Elsevier, v. 46, n. 8, p. 2067–2077, 2013.

- [45] ZHU, K. et al. No-reference video quality assessment based on artifact measurement and statistical analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, IEEE, v. 25, n. 4, p. 533–546, 2015.
- [46] XU, Q. et al. Random-valued impulse noise removal using adaptive ranked-ordered impulse detector. *Journal of Electronic Imaging*, International Society for Optics and Photonics, v. 27, n. 1, p. 013001, 2018.
- [47] MASOOD, S. et al. Statistical features based noise type identification. In: SPRINGER. *Mexican International Conference on Artificial Intelligence*. [S.l.], 2014. p. 231–241.
- [48] SANTHANAM, T.; RADHIKA, S. Probabilistic neural network—a better solution for noise classification. *Journal of Theoretical and Applied Information Technology*, v. 27, n. 1, p. 39–42, 2011.
- [49] KARIBASAPPA, K.; KARIBASAPPA, K. Ai based automated identification and estimation of noise in digital images. In: *Advances in intelligent Informatics*. [S.l.]: Springer, 2015. p. 49–60.
- [50] RODRIGUES, J.; BELO, D.; GAMBOA, H. Noise detection on ecg based on agglomerative clustering of morphological features. *Computers in biology and medicine*, Elsevier, v. 87, p. 322–334, 2017.
- [51] DEVI, M. S.; SUKUMAR, R. Metaheuristic based noise identification and image denoising using adaptive block selection based filtering. *Circuits and Systems*, Scientific Research Publishing, v. 7, n. 09, p. 2729, 2016.
- [52] RAKHSHANFAR, M.; AMER, M. A. Estimation of gaussian, poissonian–gaussian, and processed visual noise and its level function. *IEEE Transactions on Image Processing*, IEEE, v. 25, n. 9, p. 4172–4185, 2016.
- [53] YANG, S.-M.; TAI, S.-C. Fast and reliable image-noise estimation using a hybrid approach. *Journal of Electronic Imaging*, International Society for Optics and Photonics, v. 19, n. 3, p. 033007, 2010.
- [54] DONOHO, D. L. De-noising by soft-thresholding. *IEEE transactions on information theory*, IEEE, v. 41, n. 3, p. 613–627, 1995.
- [55] PYATYKH, S.; ZHENG, L.; HESSER, J. Fast noise variance estimation by principal component analysis. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Image Processing: Algorithms and Systems XI*. [S.l.], 2013. v. 8655, p. 86550K.
- [56] GUPTA, P. et al. Natural scene statistics for noise estimation. In: IEEE. *2018 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI)*. [S.l.], 2018. p. 85–88.
- [57] WEBSTER, A.; PINSON, M. *Video Quality Experts Group (VQEG)*. 2015. Disponível em: <<https://www.its.bldrdoc.gov/vqeg/vqeg-home.aspx>>. Acesso em: 30.9.2018.

- [58] BRANDAO, T.; QUELUZ, M. P. No-reference quality assessment of h. 264/avc encoded video. *IEEE Transactions on Circuits and Systems for Video Technology*, IEEE, v. 20, n. 11, p. 1437–1447, 2010.
- [59] PITREY, Y. et al. Influence of the source content and encoding configuration on the perceived quality for scalable video coding. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Human Vision and Electronic Imaging XVII*. [S.l.], 2012. v. 8291, p. 82911K.
- [60] IVP Subjective Quality Video Database. 2012. Disponível em: <<http://ivp.ee.cuhk.edu.hk/research/database/subjective/>>. Acesso em: 30.9.2018.
- [61] CHOI, L. K.; BOVIK, A. C. Perceptual flicker visibility prediction model. *Electronic Imaging*, Society for Imaging Science and Technology, v. 2016, n. 16, p. 1–6, 2016.
- [62] MOORTHY, A. K. et al. Video quality assessment on mobile devices: Subjective, behavioral and objective studies. *IEEE Journal of Selected Topics in Signal Processing*, IEEE, v. 6, n. 6, p. 652–671, 2012.
- [63] DORR, M. et al. Variability of eye movements when viewing dynamic natural scenes. *Journal of vision*, The Association for Research in Vision and Ophthalmology, v. 10, n. 10, p. 28–28, 2010.
- [64] LEDERER, S.; MÜLLER, C.; TIMMERER, C. Dynamic adaptive streaming over http dataset. In: ACM. *Proceedings of the 3rd Multimedia Systems Conference*. [S.l.], 2012. p. 89–94.
- [65] KORSHUNOV, P.; EBRAHIMI, T. Pavid: privacy evaluation video dataset. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Applications of Digital Image Processing XXXVI*. [S.l.], 2013. v. 8856, p. 88561S.
- [66] YUV Video Sequences. Disponível em: <<http://trace.eas.asu.edu/yuv/>>. Acesso em: 30.9.2018.
- [67] ABU-EL-HAIJA, S. et al. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016.
- [68] ARIAS, R.; FELINTO, A. *Youtube-8M Video List*. 2018. Disponível em: <<http://pc.cd/3egrtalK>>. Acesso em: 30.9.2018.
- [69] HALL, M. A. Correlation-based feature selection for machine learning. University of Waikato Hamilton, New Zealand, 1999.
- [70] BREIMAN, L. Random forests. *Machine learning*, Springer, v. 45, n. 1, p. 5–32, 2001.
- [71] HUBER, P. J. Robust statistics. In: *International Encyclopedia of Statistical Science*. [S.l.]: Springer, 2011. p. 1248–1251.
- [72] VLACHOS, T. Detection of blocking artifacts in compressed video. *Electronics Letters*, IET, v. 36, n. 13, p. 1106–1108, 2000.
- [73] DONG, Y.; CHAN, R. H.; XU, S. A detection statistic for random-valued impulse noise. *IEEE Transactions on Image Processing*, IEEE, v. 16, n. 4, p. 1112–1120, 2007.

- [74] SAMANTARAY, A. K. Impulse noise detection and correction by neighborhood switching median filter. In: IEEE. *High Performance Computing and Applications (ICHPCA), 2014 International Conference on*. [S.l.], 2014. p. 1–4.
- [75] BREIMAN, L. *Classification and regression trees*. [S.l.]: Routledge, 2017.
- [76] WU, T.-F.; LIN, C.-J.; WENG, R. C. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, v. 5, n. Aug, p. 975–1005, 2004.
- [77] RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *nature*, Nature Publishing Group, v. 323, n. 6088, p. 533, 1986.
- [78] FRIEDMAN, J. H. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, JSTOR, p. 1189–1232, 2001.

## TRABALHOS PUBLICADOS PELO AUTOR

Trabalhos publicados pelo autor durante o programa.

1. Rafael Lucien Bahr Arias, Alan Salvany Felinto, **Video denoising quality assessment for different noise distributions**, 2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Outubro/2018, IEEE, p. 64-71, ISBN 978-1-5386-9264-6. (Qualis CC 2017, B1)