



**Università
di Genova**

Dipartimento di
Informatica, Bioingegneria,
Robotica e Ingegneria dei Sistemi

Real-time Ultrasound Signals Processing: Denoising and Super-resolution

Simone Cammarasana

Ph.D. Thesis

Università di Genova
Dipartimento di Informatica, Bioingegneria,
Robotica ed Ingegneria dei Sistemi
Ph.D. Thesis in
Computer Science and System Engineering
Computer Science Curriculum

**Real-time Ultrasound Signals Processing:
Denoising and Super-resolution**

by

Simone Cammarasana

February 2023

Ph.D. Thesis in Computer Science and System Engineering (S.S.D. INF/01)
Dipartimento di Informatica, Bioingegneria, Robotica ed Ingegneria dei Sistemi
Università di Genova

Candidate

Simone Cammarasana
simone.cammarasana@ge.imati.cnr.it

Title

Real-time Ultrasound Signals Processing: Denoising and Super-resolution

Advisors

Paolo Nicolardi
Esaote S.p.A.
paolo.nicolardi@esaote.com
Giuseppe Patanè
IMATI, CNR
patane@ge.imati.cnr.it

External Reviewers

Nicola Vanello,
Dipartimento di Ingegneria dell'Informazione, Università di Pisa,
nicola.vanello@unipi.it
Vasileios Megalooikonomou,
Computer Engineering and Informatics Department, University of Patras,
vasilis@ceid.upatras.gr

Location

DIBRIS, Univ. di Genova
Via Opera Pia, 13
I-16145 Genova, Italy

Submitted On

February 2023

Abstract

Ultrasound acquisition is widespread in the biomedical field, due to its properties of low cost, portability, and non-invasiveness for the patient. The processing and analysis of US signals, such as images, 2D videos, and volumetric images, allows the physician to monitor the evolution of the patient's disease, and support diagnosis, and treatments (e.g., surgery). US images are affected by speckle noise, generated by the overlap of US waves. Furthermore, low-resolution images are acquired when a high acquisition frequency is applied to accurately characterise the behaviour of anatomical features that quickly change over time. Denoising and super-resolution of US signals are relevant to improve the visual evaluation of the physician and the performance and accuracy of processing methods, such as segmentation and classification. The main requirements for the processing and analysis of US signals are real-time execution, preservation of anatomical features, and reduction of artefacts.

In this context, we present a novel framework for the real-time denoising of US 2D images based on deep learning and high-performance computing, which reduces noise while preserving anatomical features in real-time execution. We extend our framework to the denoise of arbitrary US signals, such as 2D videos and 3D images, and we apply denoising algorithms that account for spatio-temporal signal properties into an image-to-image deep learning model. As a building block of this framework, we propose a novel denoising method belonging to the class of low-rank approximations, which learns and predicts the optimal thresholds of the Singular Value Decomposition. While previous denoise work compromises the computational cost and effectiveness of the method, the proposed framework achieves the results of the best denoising algorithms in terms of noise removal, anatomical feature preservation, and geometric and texture properties conservation, in a real-time execution that respects industrial constraints. The framework reduces the artefacts (e.g., blurring) and preserves the spatio-temporal consistency among frames/slices; also, it is general to the denoising algorithm, anatomical district, and noise intensity.

Then, we introduce a novel framework for the real-time reconstruction of the non-acquired scan lines through an interpolating method; a deep learning model improves

the results of the interpolation to match the target image (i.e., the high-resolution image). We improve the accuracy of the prediction of the reconstructed lines through the design of the network architecture and the loss function. In the context of signal approximation, we introduce our kernel-based sampling method for the reconstruction of 2D and 3D signals defined on regular and irregular grids, with an application to US 2D and 3D images. Our method improves previous work in terms of sampling quality, approximation accuracy, and geometry reconstruction with a slightly higher computational cost.

For both denoising and super-resolution, we evaluate the compliance with the real-time requirement of US applications in the medical domain and provide a quantitative evaluation of denoising and super-resolution methods on US and synthetic images. Finally, we discuss the role of denoising and super-resolution as pre-processing steps for segmentation and predictive analysis of breast pathologies.

Acknowledgements

I wish to thank the people who played a fundamental role during my Ph.D. studies.

My special appreciation goes to my advisors: Giuseppe Patané always encouraged me to pursue my ideas, being available for useful discussions and helping and motivating me through these years; Paolo Nicolardi supported me with his availability and professionalism and offered precious advice in the biomedical field.

I am also very grateful to my Ph.D. committee Giorgio Delzanno, Enrico Puppo, and Francesca Odone as they always provided me with valuable comments and feedback. I thank the reviewers for their thorough review and constructive comments, which helped me to improve the technical part and presentation of the Thesis. I want to thank Luigi Pampana Biancheri and Fabio Gibiino for their support and encouragement.

Finally, a special thanks to my friends and colleagues at IMATI and Unige for all the moments we shared. I enjoyed all our past and present experiences, looking forward to many other days together.

Table of Contents

| | |
|---|-----------|
| List of Figures | 5 |
| List of Tables | 11 |
| Chapter 1 Introduction | 15 |
| I Ultrasound images: background and related work | 23 |
| Chapter 2 US image acquisition and processing | 25 |
| 2.1 Ultrasound image acquisition: background | 25 |
| 2.2 US image processing: denoising and super-resolution | 30 |
| 2.2.1 US image denoising: background and related work | 31 |
| 2.2.2 US super-resolution: background and related work | 38 |
| 2.2.3 Image approximation and sampling: background and related work | 42 |
| 2.3 AI, HPC, and ultrasound imaging standards | 46 |
| 2.3.1 Artificial intelligence | 46 |
| 2.3.2 High-performance computing | 51 |
| 2.3.3 Imaging standards | 54 |
| 2.4 Experimental set-up | 57 |

| | | |
|------------------|---|------------|
| II | Real-time US denoising: contributions and novelties | 59 |
| Chapter 3 | Real-time denoising of 2D US images | 61 |
| 3.1 | Real-time denoising: proposed framework | 65 |
| 3.2 | Real-time US 2D image denoising: experimental results | 69 |
| 3.2.1 | Comparison of denoising methods | 69 |
| 3.2.2 | Framework prediction results | 71 |
| 3.3 | Learning-based low-rank denoising | 76 |
| 3.3.1 | Denoising with optimal threshold values | 77 |
| 3.3.2 | Learning model: setting and optimisation | 78 |
| 3.3.3 | Experimental results on US 2D images | 80 |
| Chapter 4 | Real-time denoising of US 2D videos and 3D images | 83 |
| 4.1 | Real-time denoising of US 2D videos and 3D images: proposed framework | 84 |
| 4.2 | Real-time denoising of US 2D videos and 3D images: experimental results | 87 |
| 4.2.1 | Comparison of denoising methods on US signals | 87 |
| 4.2.2 | Framework prediction results | 88 |
| Chapter 5 | Denoising: further experimental tests | 93 |
| 5.1 | Comparison among denoising methods on 2D images | 93 |
| 5.2 | Comparison among denoising methods on 2D videos and 3D images | 96 |
| 5.3 | Properties and tests of learning-based low-rank denoising | 98 |
| 5.3.1 | Denoising with optimal threshold values | 99 |
| 5.3.2 | Training and denoising | 100 |
| III | Real-time US super-resolution: contributions and novelties | 109 |
| Chapter 6 | Real-time super-resolution of US 2D images and videos | 111 |
| 6.1 | Real-time super-resolution framework | 113 |

| | | |
|---------------------|--|------------|
| 6.2 | Experimental results on US 2D images and videos | 117 |
| 6.2.1 | Framework prediction | 117 |
| 6.2.2 | Comparison with previous work | 122 |
| Chapter 7 | Kernel-based sampling of US and arbitrary signals | 123 |
| 7.1 | Kernel-based sampling method and application to US 2D images | 124 |
| 7.2 | Further experimental tests | 128 |
| IV | Discussion and future work | 139 |
| Chapter 8 | Conclusion and future work | 141 |
| Bibliography | | 147 |

List of Figures

| | | |
|-----|--|----|
| 2.1 | Images comparison of different acquisition methods. Images courtesy of [YTN ⁺ 16], [NEK ⁺ 20]. | 27 |
| 2.2 | Esaote ultrasound acquisition machines: MyLab X9 (left) and MyLab Omega (right). Image courtesy of [CP11]. | 28 |
| 2.3 | Ultrasound wave emission. Courtesy of [CP11]. | 28 |
| 2.4 | Example of Convolutional Neural Network. | 49 |
| 2.5 | Example of an auto-encoder network. Image courtesy of [JTJT20]. | 50 |
| 2.6 | Example of Generative Adversarial Network. Image courtesy of [FFC ⁺ 20]. | 51 |
| 2.7 | Amdahl’s Law showing the speed-up (y -axis) with respect to the number of processes (x -axis) for different fractions of sequential code. | 52 |
| 2.8 | Example of HPC architecture with CPUs, GPUs, and inter-nodes communication. Image courtesy of [WLL16]. | 53 |
| 2.9 | DICOM standard multi-document organization [Gib08]. | 56 |
| 3.1 | Pipeline of the proposed framework, with a magnification of prediction (red) and target (blue) images (right side). Our framework applies deep learning and HPC to learn and replicate the denoising results of the state-of-the-art low-rank method (i.e., tuned-WNNM), in real-time and with a specialisation to anatomic districts. | 62 |
| 3.2 | Learning-based for denoising: pipeline. | 64 |
| 3.3 | Raw images of a muscle-skeletal district and denoised images, visualised in a scan-converted format. | 65 |
| 3.4 | Raw data set of an abdominal district and denoised images, visualised in a scan-converted format. | 67 |

| | | |
|------|--|----|
| 3.5 | Raw data set of an obstetric district and denoised images, visualised in a scan-converted format. | 68 |
| 3.6 | Ultrasound image of an abdominal district and denoised images achieved by applying the tuned-WNNM and varying the denoising intensity from low (a) to high (c). | 70 |
| 3.7 | Raw, target, and prediction images, related to the obstetric data set (i). Training set: (a) 500 images, (b) 1500 images, (c) 3500 images, (d) 5000 images (Sect. 3.1). | 71 |
| 3.8 | Raw, target, and prediction images, related to the muscle-skeletal data set (ii). Training set (Sect. 3.1) with (a) 500, (b) 1500, (c) 3500, and (d) 5000 images. | 72 |
| 3.9 | PSNR and SSIM boxplot for each of the four training data set (i.e., (a-d)) and the two test data sets (Sect. 3.1): (top-left) PSNR, obstetric test data set; (top-right) SSIM, obstetric test data set; (bottom-left) PSNR, muscle-skeletal test data set; (bottom-right) SSIM, muscle-skeletal test data set. | 73 |
| 3.10 | Prediction results of the obstetric district, with the networks trained with 500 (a) and 1500 (b) images from the obstetric district, and 1500 images from multiple districts (500 obstetric, 500 cardiac, 500 muscle-skeletal images). | 74 |
| 3.11 | Prediction results of the cardiac district, with the networks trained with 500 (a) and 1500 (b) images from the obstetric district, and 1500 images from multiple districts. | 75 |
| 3.12 | (a) Input, (b) target, (c) our prediction with Pix2Pix, and (d) prediction with CNN, for the obstetric district. | 76 |
| 3.13 | (a) Minimisation of the functional (Eq. 3.2) value (y -axis) normalised on the number of vectors $N = 200$, with respect to the number of iterations (x -axis); (b) frequency (y -axis) of the difference between the optimal and the predicted thresholds (x -axis) of the first singular value of the SVD, for all the blocks of the cameraman image. | 80 |
| 3.14 | (a) Conditioning of the coefficient matrix (y -axis) with respect to N (x -axis). (b) Approximation error of the linear system (y -axis) with respect to N (x -axis). | 81 |
| 3.15 | Comparison between previous work and our denoising with low (i) and medium (ii) smoothing intensity. | 82 |
| 4.1 | Denoising of US volumetric image $312 \times 146 \times 36$ of the obstetric district. | 85 |
| 4.2 | Ray-casting [Rot82] rendering of Fig. 4.1. | 86 |
| 4.3 | Denoising of 2 frames of a 2D US video ($560 \times 359 \times 256$) of the abdomen. | 87 |

| | | |
|------|--|-----|
| 4.4 | Denoising of 2 frames of a 2D US video ($592 \times 168 \times 233$) of the cardiac district. | 88 |
| 4.5 | Denoising of 2 frames of a US video ($600 \times 380 \times 155$) of the muscle-skeletal district. | 89 |
| 4.6 | Framework prediction for 2 muscle-skeletal frames (a,b) of a 2D video. | 90 |
| 4.7 | Framework prediction for two slices (a,b) of a 3D image of the obstetric district. | 90 |
| 5.1 | Input (SIPI data set, van image), noisy (speckle noise intensity $\sigma = 0.10$), and denoised images. For error metrics, we refer the reader to Table 5.1. | 94 |
| 5.2 | Input (SIPI data set, man image), noisy (speckle noise intensity $\sigma = 0.20$), and denoised images. For error metrics, we refer the reader to Table 5.1. | 95 |
| 5.3 | Input (256×256), noisy (speckle noise intensity $\sigma = 0.05$), and denoised images with WNNM and the tuned-WNNM. | 96 |
| 5.4 | Denoising results of a phantom volumetric image with speckle noise applied: $230 \times 256 \times 80$ resolution, $\sigma = 0.05$ speckle noise. | 97 |
| 5.5 | Denoising results on a 2D video with speckle noise applied: $352 \times 288 \times 40$ resolution, $\sigma = 0.025$ speckle noise. | 98 |
| 5.6 | Input/noisy (1st row) and optimal denoising (2nd row) for: cameraman (256×256), with speckle noise and $\sigma = 0.05$; boat (256×256), salt & pepper noise with density 0.05; mandrill (256×256) Gaussian noise with $\mu = 0, \sigma = 0.01$. | 100 |
| 5.7 | 1st Row: PSNR (y -axis) comparison of the number of iterations (x -axis) between optimal (Sect. 3.3.1) values (blue), and our method's values (red). 2nd Row: average optimal thresholds (y -axis) of each singular value (x -axis). (a) Cameraman (256×256), speckle noise ($\sigma = 0.05$); (b): boat (256×256), salt & pepper noise (density 0.05); (c): mandrill, Gaussian noise ($\mu = 0, \sigma = 0.01$). Results refer to Fig. 5.8. | 101 |
| 5.8 | Comparison between previous work and our denoising: (a) cameraman (256×256), speckle noise with $\sigma = 0.05$; (b) boat (256×256), salt & pepper noise with density 0.05; (c) mandrill, Gaussian noise with $\mu = 0, \sigma = 0.01$; (d) magnification. | 102 |
| 5.9 | PSNR (y -axis) average values of our and state-of-the-art methods (x -axis) with Gaussian (a), speckle (b), salt and pepper (c) noise. | 103 |
| 5.10 | Partition of the elements (i.e., input and target vectors) for the training. Each noise intensity class (x -axis) is divided into 3 k -means classes (blue, orange, and yellow); y -axis shows the cumulated number of elements of each class. | 103 |

| | | |
|------|--|-----|
| 5.11 | (a) Input (blue), optimal (orange), and predicted (yellow) singular values (y -axis) of the cameraman image with speckle noise, for the 49 (x -axis) singular values. (b) Prediction error (y -axis) of the 49 singular values (x -axis) of a 3D block of the cameraman image with speckle noise. (c) Partition of the prediction error (y -axis), when varying the noise intensity classes (x -axis): each class is composed of three SVM-predicted classes (i.e., blue, orange, and yellow). . . . | 104 |
| 5.12 | Input/noisy (1st row) and optimal denoising (2nd row) and respective PSNR for: peppers, exponential noise with $\mu = 0.02$; wall, speckle noise with $\sigma = 0.10$; Barbara, Poisson noise; rice, Gaussian noise with $\mu = 0, \sigma^2 = 0.02$ | 105 |
| 5.13 | PSNR (y -axis) average values of our and state-of-the-art methods (x -axis) with exponential $\mu = 0.02$ (top-left), speckle $\sigma = 0.10$ (top-right), Poisson (bottom-left), Gaussian $\mu = 0, \sigma^2 = 0.02$ (bottom-right) noise. (i) BM3D, (ii) SHEAR-LAB, (iii) ACVA, (iv) DUBD, (v) WNNM, (vi) NCSR, (vii) Our denoising. . . . | 106 |
| 5.14 | Comparison between previous work and our denoising: (a) peppers, exponential noise with $\mu = 0.02$; (b) wall, speckle noise with $\sigma = 0.10$; (c) Barbara, Poisson noise; (d) rice, Gaussian noise with $\mu = 0, \sigma^2 = 0.02$ | 108 |
| 6.1 | Proposed framework: training of the learning-based model and spatial up-sampling of US videos. | 112 |
| 6.2 | Prediction on the raw images of the obstetric district: 2X up-sampling (first line); 4X up-sampling (second line). | 113 |
| 6.3 | Prediction on the raw images of the cardiac district: 2X up-sampling (first line); 4X up-sampling (second line). | 113 |
| 6.4 | Prediction on the raw images of the abdominal district: 2X up-sampling (first line); 4X up-sampling (second line). | 114 |
| 6.5 | PSNR box-plot district (left) of the (a) obstetric, (b) cardiac, and (c) abdominal districts, and error histogram (right): prediction (blue) vs. input (red): 2X (first line) and 4X (second line) results. | 115 |
| 6.6 | Concerning Figs. 6.2,6.3,6.4, we report the absolute value of the distance between the input and the prediction, for both 2X (first row) and 4X (second row) up-sampling factors. | 115 |
| 6.7 | Input and prediction of the raw images of the abdominal district 2X with different levels of brightness: low brightness (first row) and high brightness (second row). | 116 |
| 6.8 | Input and prediction of the raw images of the obstetric district 4X with different levels of brightness: low brightness (first row) and high brightness (second row). | 117 |

| | | |
|------|--|-----|
| 6.9 | Comparison of up-sampling methods vs. our method on the obstetric district: 0.5X low-resolution and 2X up-sampling. See also Table 6.1. | 118 |
| 6.10 | Comparison of up-sampling methods vs. our method on the abdominal district: 0.25X low-resolution and 4X up-sampling. See also Table 6.1. | 119 |
| 6.11 | Prediction on the denoised images of the obstetric district: 2X up-sampling (first line); 4X up-sampling (second line). | 120 |
| 6.12 | PSNR box-plot of the obstetric district with denoised images (left) and error histogram (right): prediction (blue) vs. input (red): 2X (first line) and 4X (second line) results. | 121 |
| 7.1 | (a) Input image of cardiac district (168×580), (b) image reconstruction, and (c) kernel-based sampling with 5K samples. | 124 |
| 7.2 | (a) Input image of muscle-skeletal district (500×600), (b) image reconstruction, and (c) kernel-based sampling with 5K samples (c). | 126 |
| 7.3 | (a) input MR image ($156 \times 73 \times 36$), (b) reconstruction (50K samples), and (c) different views and zoom on the sampling with 1K samples. | 127 |
| 7.4 | Variants comparison: (a), (μ)-method; (b), (σ)-method; (c), (α)-method; (d), (σ, α)-method, with 5K samples. The first column shows the input image (256×256), and the colour-map. | 128 |
| 7.5 | Accuracy of the (σ, α)-method, with a different number of samples: (a) $n = 1K$; (b) $n = 2K$; (c) $n = 4K$ | 129 |
| 7.6 | Concerning Fig. 7.4, we report the convergence of the iterative minimisation, in terms of the objective function value (y -axis) with respect to the number of iterations (x -axis). | 130 |
| 7.7 | NCC error (y -axis, right): (μ)- (red continuous line) and (σ, α)-(red dashed line) kernel-based samplings; Hausdorff distance (y -axis, left) of the corresponding samples (blue line), with respect to their number (x -axis). Image: Dante, 256×256 . | 131 |
| 7.8 | Ramp image (192×75) and sampling (1K samples) comparison among state-of-the-art methods; the percentages of each quarter show the number of samples in that portion of the grid, compared with the ink density of the input image. | 131 |
| 7.9 | Comparison of the reconstructed ramp image, between (a) [ZH16], and (b) ours. | 132 |
| 7.10 | Sampling initialisation on (a) Supermario (256×256 input points, 16K samples, $m = 4$), (b) Biosphere (512×512 input points, 50K samples, $m = 16$). | 132 |

| | | |
|------|--|-----|
| 7.11 | (a) input MR image ($128 \times 128 \times 24$), (b) reconstruction, and (c) different views and zoom on the sampling with 50K samples. | 133 |
| 7.12 | (a) input point cloud (30k points) and (b) our sampling (5k samples) of the ulna; (c) input point cloud (15k points) and (d) our sampling (3k samples) of the scaphoid | 134 |
| 8.1 | Healthcare: biomedical data management chain. | 142 |
| 8.2 | Displacement analysis of the mitral valve in the cardiac district: optical flow (1st row), optical magnitude (2nd row), magnitude magnification (3rd row). The denoising through specific algorithms allows us different characterizations of the morphological features of the anatomical part. | 143 |
| 8.3 | (a,b) Two different slices of denoising and super-resolution of Fig. 4.1 with our methods at a resolution of $624 \times 292 \times 36$; (c) Ray-tracing of the volume. | 144 |
| 8.4 | (a) US slice (400×300) of a volumetric image ($400 \times 300 \times 36$) of breast district; (b) segmentation mask; (c) extracted mesh, 15K points, 30K triangles; (d) classification of breast tumours in benign (blue) and malignant (magenta). | 145 |

List of Tables

| | | |
|-----|--|----|
| 2.1 | Comparison of different biomedical acquisition techniques. | 26 |
| 2.2 | Acoustic impedances of different body tissues and organs [CA09]. | 29 |
| 2.3 | Applications of US waves. Courtesy of [XLL ⁺ 16]. | 29 |
| 3.1 | Execution time computed as an average value of a set of Esaote images at 600×485 resolution. | 68 |
| 3.2 | Concerning the four training data sets and the two test data sets (i.e., obstetric: <i>Ob.</i> , and muscle-skeletal: <i>Msk.</i>) described in Sect. 3.1, we report the PSNR and SSIM metrics computed between the target and the prediction images, as median value among the 50 test images. | 73 |
| 3.3 | Concerning the results in Figs. 3.10, 3.11, we report the PSNR metric computed between the target and the prediction images, as the average value among the 50 images of each test data set: obstetric (<i>Ob.</i>), muscle-skeletal (<i>Msk.</i>), cardiac (<i>Card.</i>), and multiple districts (<i>Multi.</i>). The networks are trained with: single district (a, 500 obstetric images), single district (b, 1500 obstetric images) and multiple district images. | 75 |
| 4.1 | PSNR and SSIM metrics computed between the target and the prediction image, as average value among the 20 signals of each test data set: muscle-skeletal videos (<i>Msk.</i>), cardiac videos (<i>Card.</i>), obstetric 3D images (<i>Ob.</i>). | 91 |
| 4.2 | Concerning the tests in Figs. 4.1 and 4.5, we report the execution time of the denoising methods tested on 2D video and volumetric data. | 91 |
| 5.1 | PSNR and SSIM metrics of the denoising methods tested on the SIPI data set. For each σ value (i.e., the intensity of the speckle noise), we report the average metric computed on the 44 images of the data set. | 94 |

| | | |
|-----|--|-----|
| 5.2 | PSNR and SSIM metrics of WNNM and tuned-WNNM, tested on the SIPI data set. For each σ value (i.e., the intensity of the speckle noise), we report the average metric computed on the 44 images of the data set. | 96 |
| 5.3 | PSNR and SSIM metrics of the denoising methods tested on the volumetric data. Results refer to Fig. 5.4. | 99 |
| 5.4 | PSNR and SSIM metrics of the denoising methods tested on 2D video. Results refer to Fig. 5.5. | 99 |
| 5.5 | PSNR and SSIM metrics of our denoising, compared with previous work: (a) speckle, (b) salt & pepper, and (c) Gaussian noise. | 100 |
| 5.6 | With reference to Fig. 5.14, we report the PSNR metric of our denoising, compared with previous work: (a) exponential, $\mu = 0.02$, (b) speckle, $\sigma = 0.10$, (c) Poisson, and (d) Gaussian noise, $\mu = 0$, $\sigma^2 = 0.02$ | 107 |
| 6.1 | Concerning Figs. 6.9, 6.10 we report the PSNR metric computed between the target and each up-sampling method, as the mean value among the 50 test images. | 122 |
| 7.1 | With reference to Fig. 7.4, we report the approximation accuracy of the five kernel-based sampling variants, with the best results in bold. | 134 |
| 7.2 | With reference to Fig. 7.5, we report the error metrics for the (σ, α) -method with a different number of samples; best results are in bold. | 135 |
| 7.3 | With reference to Fig. 7.4, we report the increment of the execution time and the number of iterations for the five variants, with respect to the execution time $T = 24s$ (in seconds s) of the (μ) -method. | 135 |
| 7.4 | Memory allocation of the kernel-based sampling, when using $n = 5K$ samples; each variable is stored as a double precision number. | 136 |
| 7.5 | With reference to Fig. 7.5, we report the reduction of the execution time and the number of iterations for the (σ, α) -method with a different number of samples, with respect to the execution time $T = 216s$ with 5K samples. | 136 |
| 7.6 | With reference to Fig. 7.4, we report the Hausdorff distance normalised with the diagonal of the image, between the four variants (b-e) and the original kernel-based sampling (i.e., the (μ) -method). | 137 |
| 7.7 | NCC values for the reconstruction associated with the (μ) -, and (σ, α) - methods and applied to Pinocchio 256×256 | 137 |

| | | |
|-----|--|-----|
| 7.8 | With reference to Fig. 7.10, we report the approximation accuracy of the sampling optimisation. | 138 |
| 7.9 | With respect to Fig. 7.11, we report the quantitative metrics and the execution time of our method and the kernel-based sampling [ZH16]. | 138 |

Chapter 1

Introduction

In the last few decades, several techniques have been developed for the acquisition of medical images (e.g., magnetic resonance imaging, ultrasonography, elastography, positron emission tomography) and medical image processing has come into widespread use to assist diagnosis, detection, monitoring, and treatment of patients' diseases. Ultrasound (US, for short) acquisition allows physicians to visualise soft tissues (e.g., internal organs) and its application is widespread in the biomedical field, due to its low cost, portability, and non-invasiveness for the patient. High-frequency sound waves are emitted by a probe and reflected off from different layers of body tissues; the transducer converts the echoes into electrical signals that are used to create an image and display it on a screen. The image is based on the frequency and strength of the sound signal and the time the echoes took to return.

US signals have wide applications in medical specialities, e.g., analysis of muscle-skeletal pathologies, identification of tumour masses, and diagnosis of pericardial and visceral organ diseases. The processing of US signals is relevant for providing clear information to medical experts or post-processing algorithms. The main issues of US acquisition are a significant loss of information during the reconstruction of the signal, variability of the acquired data (e.g., acquisition direction, resolution, frame rate), and speckle noise that corrupts the resulting image and significantly affects the evaluation of the morphology of the anatomical district and the patient disease. Main challenges are: the processing of *large, complex* (e.g., noise, low resolution, anisotropy, distortion), and *heterogeneous* (e.g., 2D/3D images and videos) US signals of anatomical districts; the development of stable and accurate *methods for US processing* (e.g., ill-posed problem, approximation and reconstruction accuracy); and related *computational aspects* (e.g., real-time processing, memory storage).

Motivations and goals In healthcare and biomedical data management, denoising and super-resolution represent two important processing steps within the chain of operations that goes from

the acquisition of biomedical data to diagnosis. State-of-the-art *denoising* methods compromise computational costs and method effectiveness in terms of noise removal and preservation of the anatomical structures. In the context of US applications, it is relevant to develop a high-quality denoising method that respects the real-time execution constraint, which corresponds to a display frequency of 30Hz. Previous work on US images can not guarantee both the respect of the real-time constraint and the biomedical denoising characteristics of anatomical structures preservation and noise removal. State-of-the-art *super-resolution* methods, when applied to US images affected by speckle noise, tend to generate artefacts (e.g., blurring) that penalise the physician's visual assessment. The development of a real-time super-resolution method that prevents the generation of artefacts allows the probe to acquire low spatial resolution and high-frequency video, with a software-based increase of the spatial resolution.

The *overall goal* of the Thesis is the study and development of real-time denoising and super-resolution algorithms for the US images, which (i) take into account the high variability of 2D, 3D, and time-dependent US signals, (ii) are general with respect to the underlying anatomical (e.g., muscle-skeletal, abdominal, cardiac) districts, and (iii) are compliant with industrial requirements in terms of computational cost and memory requirements (e.g., on commercial US machines). Through deep learning techniques and high-performance computing, the proposed denoising and super-resolution methods are not bound to a hardware-oriented optimisation, but remain portable and limit the hardware requirements/costs for the US manufacturer. Furthermore, learning-based methods allow us to account for large biomedical data sets and specialise our methods to various anatomical districts. *Specific goals* of the Thesis are the *development of a novel deep learning framework for*

- *real-time denoising of US 2D images, and its extension to 3D US images and videos.* The proposed framework combines denoising, deep learning, and high-performance computing, and is relevant in terms of noise removal, preservation of edges of the anatomical features, and real-time computation. As a building block of this framework, we propose a novel learning-based low-rank denoising method that learns the optimal thresholds of the SVD from a training data set and applies a prediction of these optimal values to improve image denoising. The proposed method applies to US 2D images from different anatomical districts and can be inserted into our framework for the real-time denoise of US 2D images, demonstrating the generality of the framework with respect to the denoise method;
- *real-time super-resolution of US 2D images, by increasing the image resolution and reconstructing non-acquired scan lines.* Applying our approach to US videos with a low spatial resolution and a high frequency (e.g., for the cardiac district), we can generate high-frequency 2D US video with an increased spatial resolution of each frame, thus overcoming the main limits of current US probes, whose spatial resolution decreases as the acquisition frequency increases. In this context, we also introduce a novel kernel-based sampling for signals approximation and reconstruction, and discuss its application to US 2D and 3D images.

Novelties and contributions As the main contribution, the proposed deep learning framework for the denoising of US images runs in real-time and is general in terms of the type, resolution (e.g., low, isotropic, anisotropic images), dimensionality (i.e., 2D, 3D images) of the input images and the underlying anatomical district. Our approach is also general in terms of the building blocks and parameters of the deep learning framework; in fact, we can select different denoising algorithms (e.g., WNNM, SARBM3D) and deep learning architectures (e.g., Pix2Pix, VGG19). The proposed framework achieves the results of the best denoising methods in terms of noise removal, anatomical feature preservation, and geometric and texture properties conservation, in a real-time execution that respects industrial constraints.

The extension of the framework to US 3D images and 2D videos learns and replicates the denoising results of the spatio-temporal denoising in real-time with an image-to-image prediction. In this way, we achieve real-time denoising that accounts for the specific properties of the input signal and the anatomic district, thus maintaining good results in terms of denoising and edge preservation and avoiding blurring and artefacts. Then, the novel low-rank denoising trains a learning-based model to predict the optimal thresholds of the SVD. The proposed method improves previous work on 2D image denoising, and it is general with respect to noise intensity. It also provides an upper bound of the weighted nuclear norm minimisation for the denoising of 2D images and the learning results for the prediction of the optimal thresholds of the SVD. We validate our denoising framework through a quantitative comparison between predicted and target images, and a qualitative visual evaluation of the prediction by US experts. Furthermore, we perform a quantitative validation of our *versus* state-of-the-art denoising methods on synthetic signals, such as 2D/3D images and 2D videos.

We introduce a novel framework for the real-time super-resolution of US 2D images where the up-sampling is performed by an interpolating algorithm based on radial kernels, and the learning model optimises the visual accuracy of the prediction to match the high-resolution target. Our approach improves previous work on US 2D images super-resolution, with a novel network architecture in terms of the loss function and hyper-parameters. Finally, our kernel-based sampling approximates an input signal on regular (e.g., a 2D or 3D image) and irregular (e.g., surface meshes achieved by segmenting a 3D US image) domain as the sum of Gaussian kernels, whose centres, supports, and weights are computed through the minimisation of an energy functional. As the main contribution, we improve the sampling quality and the approximation accuracy and achieve more accurate feature preservation with a slightly higher computational cost.

Thesis overview and structure The Thesis is organised into four parts.

Part I describes the background and reviews previous work on the acquisition, processing, and analysis of US images and videos. In particular, we briefly review the main acquisition methodologies, such as X-ray, US, MRI, and PET, and discuss their main characteristics in terms of cost, acquisition time, and risks for the patient (*Chapter 2*). We provide an in-depth description of US acquisition, signals, and characteristics, including the acoustic properties of the US waves, the

physical properties of the probe, and the interaction of the US waves with the body structures. We describe the US signals (i.e., 2D/3D images, and 2D videos) in terms of spatial and temporal properties and how these properties affect the processing of the US signal.

Then, we introduce the characteristics required for the denoising and super-resolution of US signals, including the preservation of anatomical and geometrical structures, real-time execution, and quantitative and qualitative validation. We also discuss previous work on denoising, super-resolution, and sampling; in particular, denoising methods are classified according to the reference methodology (e.g., sparse representation, low-rank approximation) and to the dimensionality of the input signal (e.g., 2D/3D image, 2D video), with a further overview on low-rank methods. Both super-resolution and sampling related work for 2D images are classified according to related methodologies.

We describe artificial intelligence methods and their applications to biomedical problems, with an in-depth on deep learning, convolution operators, hyper-parameters, and state-of-the-art of main network architectures (e.g., CNN, GAN) that are applied to the learning-based models for image processing. We introduce high-performance computing tools in terms of properties of distributed computation, hardware, and resources, and their application to deep learning both in terms of data distribution and computation distribution. Then, we present imaging requirements in the biomedical context, with a discussion on the DICOM standard, and the data security topics that affect healthcare in terms of privacy and communication. Finally, we briefly describe the experimental test setup, in terms of hardware, software, and data set.

Part II describes the contributions and novelties for the real-time denoising of US signals. We present a novel deep learning and HPC framework (*Chapter 3*) for the real-time denoising of 2D US images, where we train a neural network to replicate the denoising results of the best state-of-the-art methods. Then, real-time denoising is achieved through the prediction of the trained network. The proposed framework combines low-rank denoising, deep learning, and high-performance computing. In the experimental tests, we perform a quantitative comparison of the prediction results compared with state-of-the-art denoising, a qualitative evaluation performed by US experts on US images acquired from different anatomical (e.g., muscle-skeletal, obstetric, and abdominal) districts, and an analysis of computational cost and execution time.

To show the generality of our approach, we propose a novel learning-based low-rank method for denoising US 2D images. Our method improves the results of previous work and can be inserted into our framework for the real-time denoise of US 2D images, thus demonstrating the generality of the framework with respect to the denoise method. Given a training data set of ground truth images, we apply artificial noise with different intensities and compute the optimal singular values through a proper optimisation applied to the SVD. The input and optimal singular values compose the training data set of the learning model, and the learning phase optimises a matrix of weights to predict the optimal thresholds of the SVD and to reconstruct the denoised image.

We extend our framework to the denoise of arbitrary US signals, such as 2D videos and 3D images (*Chapter 4*): US 2D videos allow the physician to analyse the temporal variation of an anatomical feature; 3D images offer a full understanding of the spatial anatomy of the acquired region, against a more expensive tool for the acquisition; US 2D images are simple to acquire and process, but they offer a limited view of the anatomical part of interest. Video-based and volumetric-based denoising are required for the processing of these signals since they account for the characteristics of the data (e.g., spatial/temporal redundancy, anisotropy). However, the application of these methods requires that the signal is fully acquired before its processing, thus making the denoise through these methods intrinsically non-real-time. We achieve real-time computation and maintain good results in terms of denoising and edge preservation through the extension of the framework proposed for 2D images to 3D images and 2D videos. In this application, the framework learns and replicates the dedicated denoising algorithms with an image-to-image prediction; this allows us to apply a denoising method that accounts for the properties of the specific signal, with real-time execution.

Finally, we show the validation of the denoising methods on synthetic signals, with a quantitative comparison between the ground truth and the output of the processing method (*Chapter 5*). The lack of ground truth signals is one of the main limits of the analysis of processing methods applied to biomedical data. The clinical validation and visual analysis performed by medical experts and engineers remain the main methods for assessing the quality of a processing method on medical data. However, the validation with synthetic signals allows us to quantitatively measure the accuracy of the proposed methods through specific metrics.

Part III describes the contributions and novelties of the real-time super-resolution of US signals. In particular, we present a novel framework for the real-time super-resolution of US 2D images, with an application to US 2D videos (*Chapter 6*). In the context of the dynamic acquisition, the resolution of each US image is affected by the required frequency of the video, since some anatomical districts (e.g., cardiac) require a high acquisition frequency, to accurately acquire the behaviour of anatomical features that quickly change during time.

We propose a novel framework that reconstructs the non-acquired scan lines through an interpolating method; then, a deep learning model improves the results of the interpolation to match the target image (i.e., the high-resolution image). In the experimental tests, we perform a quantitative and qualitative evaluation of our framework on a large collection of US images and videos acquired from different anatomical (e.g., muscle-skeletal, obstetric, abdominal) districts.

For signal approximation and reconstruction, we propose a novel method for the kernel-based sampling of 2D and 3D signals (*Chapter 7*). In the experimental tests, we apply our method to US 2D and 3D images, and we discuss the quantitative approximation results on synthetic 2D/3D images and 3D surfaces.

Part IV presents the discussion and future works. We summarise the results of the denoising and super-resolution methods in terms of contribution and novelties in the broader context of

healthcare data management, ranging from data acquisition to providing one or more feedback to the physician (*Chapter 8*). Then, we discuss future work, in terms of analysis and development of novel methods for the further processing of US signals, such as segmentation, morphological and predictive analysis.

Publications

Journal Papers

- **Cammarasana, Simone**, Paolo Nicolardi, and Giuseppe Patanè, “*Super-Resolution of 2D US Videos.*”, *Medical & Biological Engineering & Computing*. Under review.
- **Cammarasana, Simone**, and Giuseppe Patanè. “*Spatio-temporal analysis and comparison of 3D videos.*” *The Visual Computer* (2022): 1-16. <https://doi.org/10.1007/s00371-022-02409-1>
- **Cammarasana, Simone**, Paolo Nicolardi, and Giuseppe Patanè. “*Real-time denoising of US images based on Deep Learning.*”, *Medical & Biological Engineering & Computing* (2022): 1-16. <https://doi.org/10.1007/s11517-022-02573-5>
- **Cammarasana, Simone**, and Giuseppe Patane. “*Learning-based low-rank denoising.*”, *Signal, Image and Video Processing* (2022): 1-7. <https://doi.org/10.1007/s11760-022-02258-4>
- **Cammarasana, Simone**, and Giuseppe Patanè. “*Kernel-Based Sampling of Arbitrary Signals.*”, *Computer-Aided Design* (2021): 103103. <https://doi.org/10.1016/j.cad.2021.103103>
- **Cammarasana, Simone**, and Giuseppe Patané. “*Localised and shape-aware functions for spectral geometry processing and shape analysis: A survey & perspectives.*”, *Computers & Graphics* 97 (2021): 1-18. <https://doi.org/10.1016/j.cag.2021.03.006>

Conference Proceedings

- **Cammarasana, Simone**, Paolo Nicolardi, and Giuseppe Patanè. “*Fast Learning Framework for Denoising of US 2D Videos and 3D Images.*”, *Lecture Notes in Computer Science*, Volume 13373 (2022). https://doi.org/10.1007/978-3-031-13321-3_42
- **Cammarasana, Simone**, and Giuseppe Patanè, “*Kernel-Based Sampling of Arbitrary Data.*”, (2020) - Eurographics Italian Chapter Conference. <https://doi.org/10.2312/stag.20201252>

Extended Abstracts

- **Cammarasana, Simone**, Paolo Nicolardi, and Giuseppe Patanè, “*Signal despeckling with learned regularisation.*”, Subdivision, Geometric and Algebraic Methods, Isogeometric Analysis and Refinability in Italy (SMART), Rimini, September 2022. https://events.unibo.it/smart2022/program-and-abstracts/book_smart_2022.pdf/@download/file/book_SMART_2022.pdf
- **Cammarasana, Simone**, Paolo Nicolardi, and Giuseppe Patanè, “*A General Framework for Smoothing Arbitrary Signals in Computer Graphics and Biomedicine.*”, Arcachon, July 2022. https://cs2022.sciencesconf.org/data/pages/cs2022_abstracts.pdf

Scientific Projects

- Principal Investigator and Scientific Responsible for the ISCRA-C project “*Learning-based super-resolution of 2D US images*” (US-ML-SR) with CINECA on the Marconi100 cluster. Assigned budget: 20,000 hours;
- Principal Investigator and Scientific Responsible for the ISCRA-C project “*Denoising US Images with Spectral Methods & Deep Learning*” (IMG-DEN) with CINECA on the Marconi100 cluster. Assigned budget: 20,000 hours;
- Principal Investigator and Scientific Responsible for the ISCRA-C project “*High-Performance Computing for the Efficient Solution of PDEs on Arbitrary Domains*” (HPC-PDE) with CINECA on the Marconi cluster. Assigned budget:15,000 hours.

Posters

- **Cammarasana, Simone** and Giuseppe Patanè, “*Learning-based Low-Rank Denoising: An application to biomedical images.*”, Computer Science Workshop (CSW), DIBRIS-University of Genova, June 2022. <https://docs-dibris.github.io/assets/theme/posters/Cammarasana.pdf>

Technical Reports

- Galizia, Antonella, **Cammarasana, Simone** Clematis, Andrea, & Patane, Giuseppe. “*Evaluating Accuracy and Efficiency of HPC Solvers for Sparse Linear Systems with Applications to PDEs.*”, arXiv preprint arXiv:2201.05413 (2022). <https://doi.org/10.48550/arXiv.2201.05413>

- **Cammarasana, Simone**, Paolo Nicolardi, and Giuseppe Patanè. "A *universal Deep Learning framework for real-time denoising of US images.*", arXiv preprint arXiv:2101.09122 (2021). <https://doi.org/10.48550/arXiv.2101.09122>

Founding partners

This thesis describes the research performed as part of my three years Industrial Ph.D. project co-funded by CNR-IMATI and Esaote S.p.A. under the CNR-Confindustria agreement.

PART I

**Ultrasound images: background
and related work**

Chapter 2

US image acquisition and processing

We introduce the main methodologies for the acquisition of biomedical data (Sect. 2.1), the processing methods for ultrasound signals (Sect. 2.2), AI, HPC, and imaging standards (Sect. 2.3), and hardware, software, and tools for the experimental tests (Sect. 2.4).

2.1 Ultrasound image acquisition: background

The X-ray technique is the first acquisition method for biomedical imaging, developed in the late 19th century; the properties of the X-ray wave allow the physicians to visualise the internal structures of a patient, with the subsequent introduction of contrast agents to improve soft tissue visibility. After World War II, the introduction of the ultrasound technique allows physicians to visualise internal structures without applying ionising radiation, thus reducing the damage to human cells. US acquisition benefits many properties, such as low cost, portability, and no harm to the human body; however, ultrasound waves are less robust to perturbation than X-rays, generating noisy signals that affect the visualisation and processing of the images. In 1967, the introduction of computed tomography (CT) exploits the Radon transform to reconstruct 3D images through the collection of multiple X-ray projections. The CT requires the processing of the acquired data for image reconstruction, thus requiring more expensive hardware and longer acquisition time compared to X-rays and US. Magnetic resonance imaging (MRI) is introduced in the early 70s and applies strong magnetic fields to change the spin of atoms in human cells. The detection and processing of these changes are used to reconstruct 3D images of internal body structures. MRI does not apply ionising radiation, even if contrast agents can be applied to improve the visualisation of soft tissues. The generation of multiple magnetic fields requires specialised hardware that makes MRI an expensive acquisition method. The use of radioactively labelled markers was firstly introduced in the 50s, with the significant introduction of fluorodeoxyglucose F 18 radiotracer in 1976; the positron emission tomography (PET) accounts

Table 2.1: Comparison of different biomedical acquisition techniques.

| Exam | Cost (\$) [SM05] | Ionized radiation | Preparation time | Acquisition time |
|-------|------------------|-------------------|------------------|------------------|
| X-Ray | 410 | High | None | 10min |
| MRI | 2048 | Low | 0-30min | 10min |
| PET | 1138 | High | 60min | 30-45min |
| CT | 1565 | High | 30min | 10-20min |
| US | 410 | None | None | Real-time |

for the electron-positron annihilation events of the radiotracer, which becomes stronger in sites of high metabolic activity. The gathering of the concentration of active radioactive compounds allows physicians to visualise physiological processes, such as the metabolic process of human cells. In recent years, the joint analysis of different acquisition techniques, such as MRI-PET or CT-PET, is spreading. Instead, the ultrasound examination is applied in conjunction with CT or MRI through co-registration techniques to support surgery and patients' treatments. Table 2.1 summarises the main differences among biomedical acquisition techniques in terms of cost, acquisition time, and human body damage. Fig. 2.1 shows the different image outputs of the acquisition methods.

Ultrasound acquisition Ultrasound acquisition is applied in 1947 by Dr Karl Theo Dussik, who built the first tool to visualise the human brain and ventricles through ultrasonic waves. Nowadays, ultrasound is widespread as an imaging technology in biomedicine (Fig. 2.2). It is cheap, portable, and free of ionising radiation compared to other imaging modalities such as CT and MR. The images are acquired in real-time, thus providing instantaneous feedback for the visual interpretation of the physician and interventional procedures such as regional anaesthesia. Modern US devices apply sound waves in the range of 1–20 MHz and the acquisition is performed through a pulse-echo approach with a brightness-mode (B-mode) display: small pulses are transmitted from the ultrasound probe into the human body. The pulse is generated with two or three cycles of the same frequency, with a pulse repetition frequency (PRF) from 1 to 10kHz (Fig. 2.3). An ultrasonic wave is a longitudinal wave with a short wavelength; it propagates in any human part and is widespread thanks to its physical properties of reflection, scattering, attenuation, and Doppler effect. US waves have beam-emitting properties and propagate in one direction with strong directivity and in a straight line through the medium; the direction of the ultrasound wave along the beam is called the axial direction, and the direction perpendicular to the axial one is called the lateral direction.

The ultrasound waves penetrate the body structures along their path; when they pass through adjacent parts of the body with a different acoustic impedance, a fraction of the ultrasound pulse returns as a reflected wave, generating an echo that returns to the probe, while the rest of the wave continues to penetrate along the beam to greater tissue depths. The amplitude of the re-

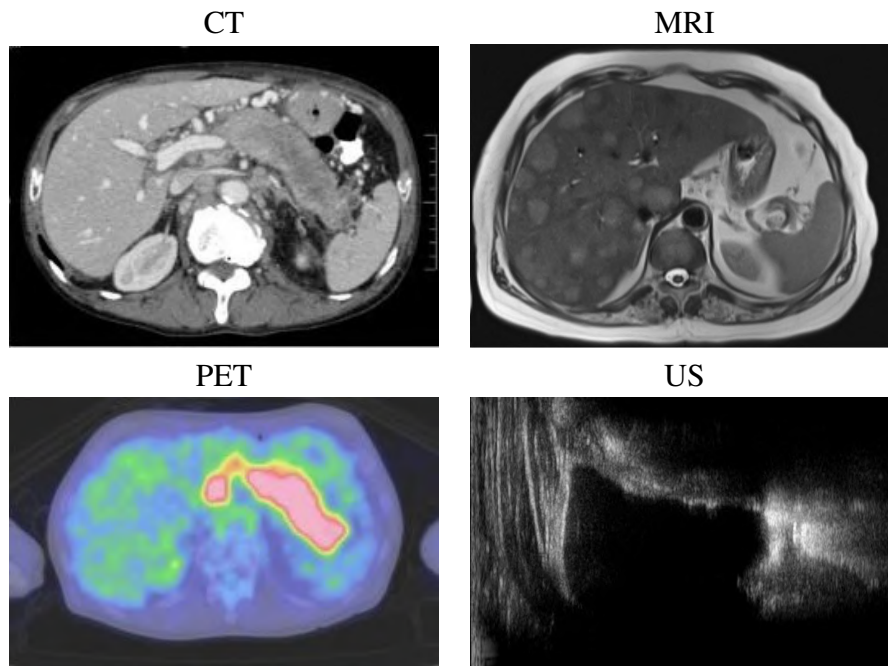


Figure 2.1: Images comparison of different acquisition methods. Images courtesy of [YTN⁺16], [NEK⁺20].

flected echo is proportional to the difference in acoustic impedance between two adjacent media. For example, interfaces between soft tissue and dense organs (e.g., bones) generate very strong echoes due to a large acoustic impedance gradient. The acoustic impedance is a physical property of a medium defined as the density of the medium times the velocity of the wave propagation. Human body tissues have different acoustic impedances (Table 2.2): for example, air-containing organs (such as the lung) have the lowest acoustic impedance, while dense organs have a higher acoustic impedance. Ultrasound transducers (or probes) contain multiple piezoelectric crystals which are interconnected electronically and vibrate in response to an applied electric current; the ultrasound probe works both as a speaker, generating the ultrasound wave, and a microphone, measuring the returned echo.

The reflected echoes are combined and processed to generate an image. High-frequency ultrasound waves (short wavelength) offer images of high axial resolution; in contrast, low-frequency waves (long wavelength) generate images of lower resolution but can penetrate deeper structures due to a lower degree of attenuation. Defining the half-value depth as the distance at which 50% of the ultrasound energy has been dissipated, 1-MHz continuous ultrasound has a half-value depth of approximately 2.5cm, while 3-MHz ultrasound has a half-value depth of approximately 0.8cm, at a fixed wave intensity of $1W/cm^2$ [DCC95]. In US images, most biologic tissues shows scattering structures. The speckle signal that provides the visible texture in organs (e.g., liver, heart) is the result of the constructive and destructive interaction of the acoustic fields be-



Figure 2.2: Esaote ultrasound acquisition machines: MyLab X9 (left) and MyLab Omega (right). Image courtesy of [CP11].

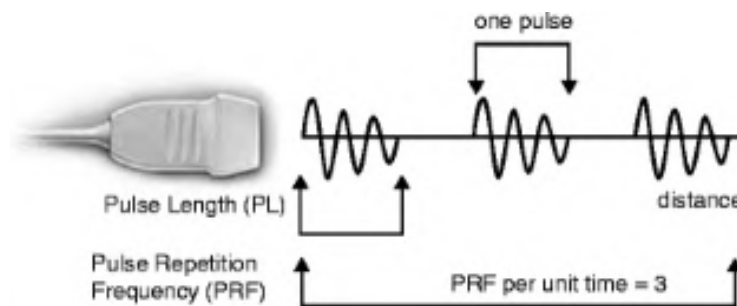


Figure 2.3: Ultrasound wave emission. Courtesy of [CP11].

tween multiple scattered echoes produced within the volume of the incident ultrasound pulse.

Finally, ultrasound waves not only penetrate the human body but also interact with body tissue. Ultrasound at low frequency and high intensity is used in the treatment of biological tissues to selectively destroy focal lesions. Table 2.3 summarises ultrasound ranges and applications.

Ultrasound signals The signal is acquired by an ultrasound machine with polar coordinates, and it is transformed into an image defined on a regular grid $m \times n$, where the number of lines m corresponds to the number of sensors of the probe and the number of columns n corresponds to the depth of the acquisition. We process US signals in their Cartesian coordinates since most of the mathematical operations are straightforward in Cartesian space; after the processing, USI is

Table 2.2: Acoustic impedances of different body tissues and organs [CA09].

| Body tissue | Acoustic impedance |
|-------------|------------------------|
| | 10×10^6 Rayls |
| Air | 0.0004 |
| Lung | 0.18 |
| Fat | 1.34 |
| Kidney | 1.63 |
| Liver | 1.65 |
| Blood | 1.65 |
| Muscle | 1.71 |
| Bone | 7.8 |

Table 2.3: Applications of US waves. Courtesy of [XLL⁺16].

| Ultrasound waves | Classifications | Applications |
|----------------------|--|--|
| Ultrasound intensity | Low-intensity ultrasound ($< 3W/cm^2$) | Therapeutic medicine, imaging medicine, medical diagnosis, and drug delivery. |
| | High-intensity ultrasound ($\geq 3W/cm^2$) | Surgery, cancer ablation, and palliative treatment. |
| Ultrasound frequency | Low frequency ultrasound (20-200 kHz) | Drug delivery, surgery, cancer ablation, and palliative treatment. |
| | Medium frequency ultrasound (0.7-3.0 MHz) | Therapeutic medicine, such as bone-fracture healing, soft-tissue lesions healing, inhibiting inflammatory responses, and erectile dysfunction treatment. |
| | High frequency ultrasound (1-20 MHz) | Imaging medicine and medical diagnosis. |

converted back into polar coordinates, for a visualisation consistent with the acquisition process.

US signals can be acquired at different planes, through the manual/automatic movement of the probe; furthermore, the physician can acquire both static and dynamic signals. These acquisitions are associated with different input signals: US images, dynamic 2D US images (i.e., videos), and 3D volumetric images. Each of these types of signals has its properties and advantages. US videos are acquired at different anatomical districts (e.g., abdominal, cardiac) and allow the physician to analyse the temporal variation of an anatomical feature (e.g., the movement of a muscle, the volume of the ventricle). This variation can be generated either by the shift of the probe or by the movement of the anatomical part. US videos are acquired through 2D probes, which capture sequences of images at a given frequency. The resolution of each frame is affected by the required frequency of the video, since some anatomical districts (e.g., cardiac) require a

high temporal detail, to better analyse the behaviour of anatomical features that quickly change during time; in particular, the relationship between image resolution and video frequency f is given by $f = c/(2 \cdot d \cdot l)$, where c is the speed of sound, l is the number of lines, and d is the depth of the acquisition.

Volumetric data are usually acquired at obstetric anatomical district through a 2D probe which is installed into an automatic tool; through the oscillation within a given range, equally spaced slices are acquired and stacked into the 3D volume; volumetric images are intrinsically anisotropic, due to the different resolution of each slice, with respect to the resolution of the probe in its movement direction. 3D volumetric images offer a full understanding of the spatial anatomy of the acquired region, against a more expensive tool for the acquisition. US 2D images are simple to acquire and process, but they offer a limited view of the anatomical part of interest. Furthermore, the processing of US 2D images is computationally cheaper than volumetric images and videos, but can not exploit the spatial and temporal redundancies of the signal.

2.2 US image processing: denoising and super-resolution

The processing of biomedical images is an interdisciplinary field, requiring medical, mathematical, informatics, physical, biological, and engineering knowledge. The application of classical computer science techniques is combined with the understanding of the acquisition tools and properties, the biological characteristics of the patient, and the requirements in terms of medical post-processing. More specifically, the processing of ultrasound images faces several challenges:

- *Anatomy* of the analysed anatomical district in terms of structures, geometries and textures is relevant to evaluate the quality of the applied processing method;
- *Instrumentation* affects the output signal in terms of noise, resolution, isotropy, and artefacts; the knowledge of the physical properties of the acquisition tools allows us to optimise the application of the processing methods;
- *Acquisition modality*, such as manual vs. automatic, static vs. dynamic, affects the output signal in terms of artefacts and processing requirements. The processing of different signals (e.g., 2D vs. 3D images) requires methods that account for the intrinsic properties of the signal;
- the *clinical evaluation* is a fundamental step for the evaluation of the performance of the processing methods due to the lack of biomedical ground truth references; for this reason, the qualitative validation of experts of US images allows us to evaluate the results in terms of preservation of the anatomical features;

- *Industrial requirements* set boundaries to the processing algorithms. The main industrial and clinical constraint in the pre-processing of ultrasound signals is real-time calculation. The physician must be able to see in real-time the changes in the signal as the probe moves. In particular, the required constraint is about 30 milliseconds per image, which corresponds to a frame-per-second resolution of 30Hz. This result must be guaranteed on current ultrasonic machines equipped with modern CPUs and GPUs. We test the denoising prediction on GPU-based hardware, which replicates the hardware of an ultrasound scanner currently in use.
- The *medical application* in diagnosis or intervention provides the foundation and motivation for USI analysis; the processing depends on many medical factors, such as the suspected disease, the anatomical district, and the planned post-processing/clinical steps.

In this context, we discuss background and related work for US image denoising (Sect. 2.2.1), super-resolution (Sect. 2.2.2), and approximation and sampling (Sect. 2.2.3).

2.2.1 US image denoising: background and related work

Ultrasound acquisition is a widespread technology in medical diagnosis, with several benefits in terms of cost, portability, and non-invasiveness. The main issues of the ultrasound technique are a significant loss of information during the reconstruction of the signal, the dependency of the signal from the direction of acquisition, and an underlying noise that corrupts the image and significantly affects the evaluation of the morphology of the anatomical district. In particular, speckle noise is a multiplicative noise that is generated when the scale of organ tissue's structures is close to the ultrasonic wavelengths; different waves are reflected back and overlap generating the perturbation in the signals that appears in the output image as darker and brighter grains. Speckle noise affects the evaluation of ultrasound images, resulting in poor image contrast degree, inadequate and non-identifiable characteristics of the tissue and structure features, and making it more difficult to identify geometries and edges of anatomical features, thus influencing the accuracy of diagnoses. In this context, denoising is relevant and widespread in US processing, and many works propose various appropriate denoising methods to address the specific features of ultrasonic images (Sect. 2.2.1). Ultrasonic image processing through denoising methods has pros and cons: noise reduction improves visual evaluation and the application of post-processing methods that use local information and are therefore influenced by noisy elements that alter the accuracy of numerical operators, such as convolution and interpolation. At the same time, applying a denoising algorithm means altering the image and therefore potentially the contours and anatomical structures. Additionally, image denoising increases the time from data acquisition to screen display for the physician. In this context, it is relevant to consider denoising as an additional tool for the visualisation of biomedical information, which does not delete the image affected by noise but accompanies it. Possibly joint analysis of the two im-

ages (i.e., noisy and denoised) allows the physician to have a complete picture and improve the diagnosis. For these reasons, a valid denoising algorithm must satisfy the following properties:

- *smoothing* of pixels with high-gradient features;
- *preservation* of anatomical structures, edges, geometries, and texture properties;
- *reduction* of artefacts and blurring effects;
- *computational cost* compatible with real-time visualisation constraint;
- *memory consumption* compatible with hardware requirement;
- *versatility* with different anatomical districts;
- *adaptability* to the visual perception of a different physician, through the proposal of soft and hard smoothing;
- *qualitative and clinical validation* by an industrial or medical expert, due to the lack of ground truth references for the evaluation of the denoising properties;
- *signal-driven* denoising, taking into account the spatio-temporal properties of the input data, including 2D and 3D images, and 2D videos;
- *hardware* requirements, in terms of CPUs and GPUs computing power.

Denoising of ultrasound images is relevant both for post-processing and visual evaluation by medical experts. Despite some relevant works consider raw ultrasound images and videos for cardiac segmentation [LFL⁺21, OHG⁺20], several works show the benefits of denoising for *segmentation* [YQX⁺12, TGS⁺10, ZLRQ19], *feature extraction* [IKM08], *classification* [WDW⁺20, SJ21], *super-resolution* [KAR18], *registration* [DSFC⁺13], and *texture analysis* [PPBN05]. Furthermore, main ultrasound machine manufacturers include a denoising filter in their scanners [urlb, urld]. In terms of hardware requirements, several ultrasound machines manufactured by main competitors (e.g., Esaote, Philips) are equipped with GPU cards [urlb, urlc]. Furthermore, some recent denoising methods for ultrasound images are developed on GPUs [PXdFABCH11, BSD18]. Finally, the application of GPUs to image processing for future medical ultrasound imaging systems [SCYY11] presents the advantages of GPUs over CPUs in terms of performance, power consumption, and cost.

US denoising: related work

We review previous work on the denoise of 2D/3D images and 2D videos.

Non-local methods The *Non-Local Means* (NLM) denoising [BCM05] uses the patterns' redundancy of the input image, and each patch is restored with a weighted average of all the other patches, where each weight is proportional to the similarity among the patches. The *Bayesian non-local mean filter* [KBC07] improves the NLM with the introduction of a Bayesian estimator as a distance measure among the patches, which allows the user to better determine the amount of denoising by the noise variance of the patch. The anisotropic neighbourhood in NLM [MNB13] uses an image gradient to estimate the edge orientation and then adapts the patches to match the local edges. The characterisation of the patches through a redundancy index [MZY20] improves the self-similarity computation among patches. The improvement on the structure of the search window is achieved through the computation of an optimal search window for each pixel [VP17], according to the denoising degree of the related patch.

Anisotropic methods The denoised image is computed as the solution to an anisotropic diffusion equation [PM90, Pat15], where the gradient of the image guides the diffusion process. The variant [YA02] exploits the Lee [Lee80] and Forst [FSSH82] filters, which are edge-sensitive to speckle noise. An improvement of the previous results [AFAL06] is achieved by applying the Kuan filter [KSSC85] in the diffusion equation and revising the selection of the neighbourhood used for the estimation of the statistical parameters. The anisotropic method introduces a class of fractional-order anisotropic diffusion equations [BF07], using the Fourier transform to compute the fractional derivatives, and the discrete Fourier transform to compute the fractional-order differences.

Spectral denoising Denoising based on *spectral decomposition* transforms a signal into its spectral domain and exploits the sparsity of the transformed signal to remove noise through a threshold operation. Several transformations have been applied to image denoising, such as Wavelets [MKRM99, CYV00, PSWS03, LZC⁺17], Curvelets [SCD02], Contourlets [DCZD06], and Shearlets [YWNL14]. To reconstruct the denoised image, the *3D block-matching* [DFKE06] computes and stacks similar patches through NLM; each stack is transformed into its spectral domain with wavelet decomposition, denoised through a hard/soft threshold, and reconstructed in the space domain. The denoised patches are aggregated by a collaborative filter. The *synthetic aperture radar block matching 3D* (SAR-BM3D) [PPAV11] introduces a speckle-based variant of 3D block matching; the similarity among the patches is computed by considering the probability distribution of the speckle noise as a distance metric. Furthermore, the hard/soft threshold of the wavelet transformed signal is substituted by a Local Linear Minimum Mean Square Error (LLMMSE) filter. The *principal component analysis block matching 3D* (PCA-BM3D) [DFKE09] improves the stacking operation of 3D block-matching by using shape-adaptive neighbourhoods, which enables its local adaptability to image features. The 3D transformation of each stack to the spectral domain is performed through the PCA [WEG87] and an orthogonal 1D transformation in the third dimension.

Low-rank methods Low-rank approximation computes the denoised image as the solution to a weighted minimisation problem, whose cost function is the Frobenius norm [SJ03, CNP21, Pat15] or the ℓ_1 norm [EVDH10], between the input and the target images. The relation between local and non-local information [DSL12] allows us to estimate signal variances, by interpreting the *Singular Value Decomposition* (SVD, for short) through a bilateral variance estimation. In [RRB12], a high-order SVD is applied to 3D blocks, and the denoised image is achieved with hard thresholding of the decomposed signal. The *Weighted Nuclear Norm Minimisation* (WNNM) [GZZF14] computes the stacks as in the 3D block-matching method, performs the SVD on the stacks and applies a weighted threshold to the singular values, where higher weights correspond to lower singular values, which capture the noisy component of the image. The *collaborative filtering* of WNNM for the aggregation of the denoised patches is performed as in the 3D block-matching method. The weighted nuclear norm and the histogram preservation [ZD18] are combined in a single constrained optimization problem, which is solved through the alternating direction method of multipliers [BPC⁺11]. The WNNM is extended to image deblurring with several types of noise [MXZ17].

External learning A *learned simultaneous sparse coding* method [MBP⁺09] integrates sparse dictionary learning with non-local self-similarities of natural images. The *non-locally centralised sparse representation* (NCSR) [DZSL12] exploits the non-local redundancies, combined with local sparsity properties, to estimate the coefficients of the sparse representation of the input image. The dictionary is learned by clustering the patches of the image into K clusters through the K-means [Mac67] method and then learning a PCA sub-dictionary for each cluster. This method has been further improved in [XYJ17] with a fast version based on a pre-learned dictionary and achieving an improvement of computational efficiency. The structured sparse model selection over a family of learned orthogonal bases [MZ16] is applied to the deblurring of images with Gaussian noise.

Deep learning methods for denoising In the *Noise2Noise* algorithm [LMH⁺18], the network learns to denoise images only considering the noisy data, without any knowledge of the ground truth. The *Noise2Void* algorithm [KBJ19] further expands this idea, and it does not require couple of noisy images for the training. This approach is relevant in biomedical fields, where there are no ground truth images. The *Noise2Self* method [BR19] proposes a self-supervised algorithm that does not require any prior information on the input image, estimation of the noise, or ground truth data. The denoising of images [FZ20] is achieved through the extraction of features from the noisy image through a convolutional neural network (CNN) and combining the edge regularisation with the total variation regularisation. The combination of CNN and low-rank representation [FJZ⁺21] is applied to detect anomalous pixels in hyperspectral images. The multilevel wavelet convolutional neural network [WLJ⁺20] is applied for restoring blurred images affected by Cauchy noise. The *block matching Convolutional Neural Network* (BM-CNN) [AC17] integrates a deep learning approach with the 3D block-matching method; the

denoising of the stacks is predicted through a DnCNN [ZZC⁺17], which is trained with a data set of 400 images corresponding to more than 250K training samples. A feed-forward Convolutional Neural Network smooths images, independently from the noise level, by exploiting residual learning and batch normalisation. Then, the blocks are aggregated and the image is reconstructed, as in the 3D block-matching algorithm.

Denoising of 2D US videos and 3D images The extension *Block Matching 4D* (BM4D) [MKEF12] of the BM3D algorithm to volumetric images exploits grouping and collaborative filters, where similar voxels are stacked together into 4D groups. A variant of non-local means denoising to 3D biomedical images [MCMB⁺10] takes into account the spatially varying noise levels across the volumetric data, and automatically adjusts the strength of the filter through a noise estimation approach. The K-SVD algorithm [AEB06] represents the signal as a sparse linear combination of atoms from an over-complete dictionary; the atoms are iteratively updated through an SVD of the representation error to better fit the data. A 3D blockwise version of the non-local means filter with wavelet sub-bands [CHP⁺08] is applied to biomedical images. A learning-based method is applied to denoise 3D biomedical data [RHC⁺19] through a residual encoder-decoder Wasserstein generative adversarial network.

Vidosat [WRB18] applies online transform learning to 2D video denoising; the patches are computed either from corresponding 2D patches in successive frames or using an online block matching technique, fully exploiting the spatio-temporal data correlation. In VBM3D [MBFE12], 3D spatio-temporal volumes are constructed by tracking blocks along trajectories, which are defined by the motion vectors. Then, the denoising of the blocks is achieved through collaborative filtering, by transforming each block through a decorrelating 4-D separable transform; finally, the shrinkage and inverse transformation generate the denoised output. The formulation of the denoising of mixed noise as a low-rank matrix completion problem [JLSX10] leads to a denoising scheme without strong assumptions on the statistical properties of the noise. The application of the non-separable oriented 3-D dual-tree wavelet transformation [SL03] gives a motion-based multi-scale decomposition for video and isolates in its sub-bands motion along with different directions.

Among state-of-the-art methods, we select three denoising methods: 2D-based method (e.g., *Weighted Nuclear Norm Minimisation* - WNNM [GZZF14]), 3D-based method (BM4D), and Vidosat. In particular, 2D and volumetric methods (i.e., WNNM, BM4D) account for the spatial relationship among patches, to exploit data similarity and remove the noise component. In contrast, video denoising (e.g., Vidosat) exploits both the spatial and the temporal dimensions of the data.

Low-rank approximation problems

Given a noisy signal \mathbf{Y} , we want to estimate the ground truth signal \mathbf{X} with respect to the noisy component \mathbf{N} , as $\mathbf{Y} = f(\mathbf{X}, \mathbf{N})$ where f is a generic operator (e.g., additive, multiplicative). Both the signals and the noise can be defined as 2D matrices $m \times n$ or as continuous functions $\mathbf{X}, \mathbf{Y}, \mathbf{N} : \Omega \subset \mathcal{R}^2 \rightarrow \mathcal{R}$. In this context, a denoising method aims at finding an approximation of the noisy signal that penalises undesirable solutions through a form of regularisation

$$\hat{\mathbf{X}} = \arg \min_{\mathbf{X}} \{ \|\mathbf{Y} - \mathbf{X}\|^2 + \lambda P(\mathbf{X}) \}, \quad (2.1)$$

where $\|\mathbf{Y} - \mathbf{X}\|^2$ is the discrete sum/integral over the domain of the squared difference of the noisy and ground truth signal, in the discrete and continuous context respectively. The $P(\mathbf{X})$ denotes a penalty function which is λ -weighted and is higher as the solution recedes the target one, according to the desired properties. In this context, several formulations describe different classes of representation and regularisation.

In sparse-representation models (e.g., Fourier, Wavelet), the predicted signal is represented as a sparse representation Θ of a dictionary of basis functions \mathbf{W} . The minimisation problem (Eq. 2.1) finds the minimum number of basis to represent the noisy signal and the penalisation expresses certain properties of the dictionary-based representation, e.g., the reduction of high-frequency components of the Fourier transformation, as

$$\hat{\Theta} = \arg \min_{\Theta} \{ \|\mathbf{Y} - \mathbf{W}\Theta\|^2 + \lambda P(\Theta) \}.$$

Total variation method is a variational model

$$\hat{\mathbf{X}} = \arg \max_{\mathbf{X}} \left\{ \int_{\Omega} (\mathbf{Y}(\mathbf{t}) - \mathbf{X}(\mathbf{t}))^2 + \lambda P(\mathbf{X}) \right\}$$

where the penalisation is the integral of the norm of the gradient of the solution $P(\mathbf{X}) = \int_{\Omega} |\nabla \mathbf{X}(\mathbf{t})| dt$

The nuclear norm minimisation (NNM) model penalises the nuclear norm, which is defined as the sum of the singular values of a matrix $\|\mathbf{X}\|_* = \sum_i |s_i(\mathbf{X})|$ as

$$\hat{\mathbf{X}} = \arg \min_{\mathbf{X}} \{ \|\mathbf{Y} - \mathbf{X}\|^2 + \lambda \|\mathbf{X}\|_* \}.$$

NNM can be equivalently solved as a low-rank problem by soft-threshold the singular values of the variable matrix through a fixed quantity λ , as shown in [CCS10]. The NNM is further generalised to the Weighted Nuclear Norm Minimisation (WNNM, [GZZF14]), where different threshold values are applied to each singular value, as

$$\hat{\mathbf{X}} = \arg \min_{\mathbf{X}} \left\{ \|\mathbf{Y} - \mathbf{X}\|^2 + \|\mathbf{X}\|_{w,*} \right\}, \quad (2.2)$$

where $\|\mathbf{X}\|_{w,*} = \sum_i |w_i s_i(\mathbf{X})|$, with $w_i \geq 0$ is a non-negative weight assigned to $s_i(\mathbf{X})$. In this context, we discuss some numerical and computational properties of the singular values decomposition (SVD) of a matrix.

SVD: numerical properties Every matrix $\mathbf{A} \in \mathcal{C}^{m \times n}$ has a singular values decomposition, which is the factorisation of \mathbf{A} into the product of three matrices $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$. \mathbf{U} is $m \times m$ orthogonal matrix, whose columns u_1, \dots, u_m are m -dimensional orthonormal left singular vectors of \mathbf{A} ; \mathbf{V} is $n \times n$ orthogonal matrix, whose columns v_1, \dots, v_n are n -dimensional orthonormal right singular vectors of \mathbf{A} ; \mathbf{S} is an $m \times n$ diagonal matrix with non-negative entries sorted in decrescent order. The singular values are uniquely defined; if the singular values are all distinct, then the right and left singular vectors are unique. On the contrary, when some sets of singular values are equal, the corresponding singular vectors span some subspace and any set of orthonormal vectors spanning this subspace can be used as the singular vectors.

The columns and the rows of \mathbf{U} and \mathbf{V} are computed as the eigenvectors of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$ respectively, while the diagonal elements s_i of \mathbf{S} are the square roots of the n eigenvalues of $\mathbf{A}^T\mathbf{A}$, with $s_1 \geq s_2 \geq \dots \geq s_n$. Moreover, the 2-norm and the Frobenius norm of \mathbf{A} are $\|\mathbf{A}\|_2 = s_1$ and $\|\mathbf{A}\|_F = \sqrt{s_1^2 + \dots + s_r^2}$ respectively, where r is the rank of \mathbf{A} . The image space and the null space of \mathbf{A} can be computed as $\text{range}(\mathbf{A}) = \langle \mathbf{u}_1, \dots, \mathbf{u}_r \rangle$ and $\text{null}(\mathbf{A}) = \langle \mathbf{v}_{r+1}, \dots, \mathbf{v}_n \rangle$. Geometrically, a $m \times n$ matrix is an application from \mathcal{R}^n to \mathcal{R}^m that performs a rotation in the domain (multiplication by \mathbf{V}^T), followed by scaling plus adding or deleting dimensions (multiplication by \mathbf{S}), followed by a rotation in the range (multiplication by \mathbf{U}).

The matrix \mathbf{A} of rank r can be approximated through a low-rank matrix of rank k , as $\mathbf{A}_k = \sum_{i=1}^k s_i \cdot \mathbf{u}_i \mathbf{v}_i^T$. This is equivalent to $\mathbf{A}_k = \mathbf{U}_k \mathbf{S}_k \mathbf{V}_k^T$, where \mathbf{U}_k and \mathbf{V}_k are the first k columns of \mathbf{U} and \mathbf{V} respectively, and \mathbf{S}_k are the first k singular values, i.e., the first k elements of the diagonal of \mathbf{S} . \mathbf{A}_k is the best approximation of rank k of the matrix \mathbf{A} . In fact, for every $m \times n$ matrix \mathbf{A} , rank target $k \geq 1$, and $m \times n$ matrix \mathbf{B} of rank k , $\|\mathbf{A} - \mathbf{A}_k\|_F \leq \|\mathbf{A} - \mathbf{B}\|_F$, where \mathbf{A}_k is the rank- k approximation derived from the SVD of \mathbf{A} . Furthermore, we estimate the approximation error of \mathbf{A}_k with respect to \mathbf{A} as $\|\mathbf{A} - \mathbf{A}_k\|_2 = s_{k+1}$ and $\|\mathbf{A} - \mathbf{A}_k\|_F = \sqrt{s_{k+1}^2 + \dots + s_r^2}$. The low-rank approximation \mathbf{A}_k requires $\mathcal{O}(k(m+n))$ instead of $\mathcal{O}(m \cdot n)$ memory. Finally, the solution of Eq. 2.2 can be computed through the weighted threshold of the SVD decomposition as $\hat{\mathbf{X}} = \mathbf{U}\hat{\mathbf{S}}\mathbf{V}$, where

$$\hat{\mathbf{S}} = \arg \min_{\mathbf{S}^*} \left\{ \|\mathbf{S} - \mathbf{S}^*\|_F^2 + \|\mathbf{S}^*\|_{w,*} \right\}.$$

SVD: computational properties Most of the algorithms developed for the computation of the SVD find square roots of eigenvalues of $\mathbf{A}^T\mathbf{A}$ without actually computing the matrix-matrix multiplication, which is a computationally expensive operation. The reduction of the input matrix to a bidiagonal form through the Householder transformation [Hou58] allows several algorithms to

compute the SVD through bisection [DK90], divide and conquer [GE95], and QR factorisation [Dem97] methods. Other algorithms directly apply on the input matrix through diagonalising rotations, which are a typology of orthogonal transformation that keeps track of a matrix’s eigenvalues, eigenvectors, singular values, and singular vectors as well as angles and lengths. The application of a sequence of rotations \mathbf{A}^k converges to the diagonal matrix of the singular values, i.e., $\lim_{k \rightarrow \infty} \mathbf{A}^k = \mathbf{S}$; in particular, the Jacobi rotation annihilates a symmetric pair of off-diagonal entries. In [Kog55], the analogous of the Jacobi method for symmetric matrices is applied. The application of the cyclic Jacobi method [FH60] consists of an iteration of sweeps which does not require the determination of the largest off-diagonal element of \mathbf{A}^k and an orthogonal reduction of the $m \times n$ input matrix generates an output matrix where all the non-zeros are in the upper $n \times n$ portion. The *Hestens-Jacobi* method [Hes58] accounts for the equivalence between the orthogonalisation of two vectors and the annihilation of a matrix element by means of orthogonal plane rotations.

Recently, the development of methods for the SVD accounts the parallel computation, favouring those algorithms that allow a distribution of the operations among different processes. In [SHA03], a small increase in the computational cost of the method by sacrificing the convergence rate allows the algorithm to distribute the calculation over several processes with benefits in terms of computational cost. For further details on the SVD decomposition and its application for low-rank problems, we refer the readers to [RV15], [CD06] and [KL80].

2.2.2 US super-resolution: background and related work

Through US videos, the physician analyses the temporal variation of an anatomical feature (e.g., the movement of a muscle, the volume of the ventricle), which can be generated either by the shift of the probe or by the movement of the anatomical part. 2D US videos are acquired through 2D probes, which capture sequences of images at a given frequency. The resolution of each image is affected by the required frequency of the video, since some anatomical districts (e.g., cardiac) require a high acquisition frequency, to accurately acquire the behaviour of anatomical features that quickly change over time. The underlying image has a resolution of $l \times d$, where l is the number of scan lines (i.e., the lateral resolution), and d is the depth of the acquisition of each scan line (i.e., the axial resolution). Axial resolution refers to the ability to discern two separate objects that are longitudinally adjacent to each other in the ultrasound image; lateral resolution refers to the ability to discern two separate objects that are adjacent to each other; the lateral resolution is usually lower than axial resolution in ultrasound. The resolution of the image in terms of lateral direction (i.e., the direction perpendicular to the US propagation along the beam line) is primarily determined by the width of the ultrasound beam and the number of elements (i.e., the piezoelectric crystals) that are activated to generate the US waves. Current probes vary the number of beam-lines acquired by activating/deactivating piezoelectric crystals thus reducing lateral resolution and image acquisition time. The axial resolution can be varied by changing the

length and frequency of the pulses, which affect the penetration of the ultrasound wave. In this context, we focus on lateral low-resolution image acquisition, reduce the acquisition time, and subsequently reconstruct the high-resolution image without losing information in terms of data depth. The US image super-resolution is relevant to improve the quality of the image, its visual interpretation by the physician, and post-processing steps, e.g., classification [ANMM⁺17], diagnosis [BSGR⁺21], and segmentation [LZT⁺22].

The spatial resolution of an ultrasound imaging system is limited by the diffraction limit to length scales of approximately half of the wavelength of the transmitted beam; two or more point reflectors that are spaced more closely than this limit can not be distinguished. We mention that super-resolution microscopy [HW94] is a series of methods in optical and acoustic microscopy that apply fluorescence photo-activated localisation to break the diffraction limit of the sound wave and visualise microvascular structures in the order of tens of micrometres. In this context, the super-resolution microscopy is applied for the segmentation [BGH20], co-registration [ATK⁺21], and morphological analysis [SZA⁺21] of anatomical structures.

Finally, ultrasound images are affected by a significant speckle noise, which in addition to worsening the visual interpretation by the physician, makes it more difficult to apply the classic super-resolution methods, based on interpolation and/or deep learning. In fact, the presence of noise can lead to an approximation of the reconstructed grey intensities that is incorrect and far from the real values. In this context, the required characteristics of the super-resolution method are:

- *Features preservation*, including the edges and geometries of organs and tissues;
- *Artefacts reduction* in terms of blurring and shading;
- *Real-time* processing, due to the industrial and clinical requirements of USI processing methods;
- *Up-sampling factors*, the super-resolution methods apply to lateral resolution only, with up-sampling factors of 2X and 4X;
- *Anatomical districts*, with specialised methods for different anatomical organs, e.g., cardiac, muscle-skeletal;
- *Input signal*, 2D and 3D images, 2D videos.

US image super-resolution: related work

We review previous work on super-resolution for both generic and US images.

Deep learning super-resolution of generic images In the last years, deep learning methods for super-resolution are widespread. A sparse representation of patch pairs over two dictionaries, one for both low-resolution (LR) and high-resolution (HR) images [PE14] captures the statistical dependencies between the sparsity patterns of the low and high-resolution coefficients of the corresponding representations. In [DLHT14, DLHT15]. A fully convolutional neural network exploits a large filter size in the non-linear mapping layer and accounts for three colour channels to train and predict HR images from LR images. A data clustering, which groups patches and a Bayes strategy for patch selection [SPP15] provides a fast up-sampling based on external learning. The main novelties of the enhanced deep super-resolution network [LSK⁺17] are a simplification of the conventional residual network architectures and a multi-scale super-resolution network that reduces the model size. A *super-resolution generative adversarial network* (SRGAN) [LTH⁺17] applies a deep residual network with skip-connection and a perceptual loss between generated and target images. The reduction of artefacts of the previous method is addressed by the Enhanced SRGAN [WYW⁺18], which improves the network architecture, the adversarial and the perceptual loss, removes the batch normalisation layer, and applies the residual scaling and smaller initialisation values. The perceptual quality of ESRGAN is improved by the ESRGAN+ method [RR20] through a novel *Residual-in-Residual Dense Residual* block, which increases the network capacity without affecting its complexity. The introduction of weight normalisation and wider features before rectified linear unit activation function [YFH20] achieves good up-sampling results, with a low computational cost. Additional methods, classified according to supervised/unsupervised approach, and domain-specific applications, are discussed in [WCH20].

Vision-based super-resolution of generic images The interpolating up-sampling with cubic kernels [Key81] offers high accuracy with low computational cost, through appropriate boundary conditions and constraints on the kernel functions. The interpolated values are computed as weighted average of pixels in the 2X2 (bilinear, [GJB03]) or 4X4 (bi-cubic, [MMP⁺14]) neighbourhood. A fast implementation of bilinear and bi-cubic interpolations [KAJ⁺20] is applied to *Field Programmable Gate Array* (FPGA), reducing the computational complexity and the FPGA resources while providing an excellent trade-off between image quality and calculation simplicity. After the training of two dictionaries for LR and HR patches [YWHM10], the similarity of the sparse representation of LR and HR patches with the respective dictionary is exploited to generate the high-resolution image. Anchored neighbourhood regression [TDSVG13] and its improved version [TDSVG14] learn a regression to correlate LR and HR images for each atom of the dictionary and precomputed neighbourhood. The search of recursive patches within an image [HSA15] is extended by allowing geometric variations, incorporating the geometry by localising planes, and accounting plane parameters to estimate the deformation of recurring patches. Bivariate rational fractal interpolation [ZFB⁺18] improves the approximation results with respect to polynomial kernels, preserving image edges and textures.

Learning-based US super-resolution The main novelties of the enhanced deep super-resolution network [LSK⁺17] are a simplification of the conventional residual network architectures and a multi-scale super-resolution network that reduces the model size. Exploiting the sparsity of the signal in the Fourier domain, the interpolation of missing data [YY18] allows reconstructing the HR US image with a low computational cost. A *super-resolution generative adversarial network* (SRGAN) [LTH⁺17] applies a deep residual network with skip-connection and a perceptual loss between generated and target images. The reduction of artefacts of the previous method is addressed by the Enhanced SRGAN [WYW⁺18], which improves the network architecture, the adversarial and the perceptual loss, removes the batch normalisation layer, and applies the residual scaling and smaller initialisation values. The perceptual quality of ESRGAN is improved by the ESRGAN+ method [RR20] through a novel *Residual-in-Residual Dense Residual* block, which increases the network capacity without affecting its complexity. The application of the SRGAN to US images [CKH⁺18] preserves both the anatomical structures and the speckle noise pattern, thus improving the perceptual quality of the upsampled images. Dilated convolution [LL18] extracts the internal recurrence information from the test image; this method upsamples LR images when LR-HR examples are reduced. Fully convolutional U-net [VSSB⁺19] obtains high-resolution vascular images from high-density contrast-enhanced US signals. In [TB20], the deep learning method exploiting feature extraction blocks, repeating blocks, and upsampling layers apply an up-sampling factor in the range 2-8. A Self-supervised CycleGAN [LLH⁺21] only requires the LR US image, and generates perceptually consistent up-sampling results. Combining CycleGAN, two-stage GAN, and the zero-shot super resolution [DZTN21], it is possible to obtain super-resolution images with low blurring artefacts.

Vision-based US super-resolution Learning-based methods suffer from artefacts and blurring when dealing with noisy signals. Several vision-based methods have been proposed, through the years. The interpolating up-sampling with cubic kernels [Key81] offers high accuracy with low computational cost, through appropriate boundary conditions and constraints on the kernel functions. In [AMP⁺11], a novel deconvolution-based method applies the maximum a posteriori estimation to the restoration of the tissue response and is validated with several tissue-mimicking phantoms with specific scatterer concentrations. The *Alternating Direction Method of Multipliers* [NWY10] is applied to the super-resolution of US images including deblurring and denoising [MBK12] through a combination of ℓ_1 and ℓ_2 minimisation. In [YZX12], a deconvolution method models the envelope radio-frequency and point spread function is robust to noise and does not require the knowledge of the centre frequency of the acquired signal. Assuming a Gaussian distribution for both the unknown signal to be restored and the point spread function, in [ZBKT15] the reconstructed image is built through a posterior model with hybrid Gibbs sampling [GG84]. The properties of the decimation matrix in the Fourier domain [ZWB⁺16] are exploited to solve the super-resolution problem with a ℓ_p -norm regulariser, with $p \in [1, 2]$. The envelope of radio frequency signal [KAR18] applies repetitive data in the non-local neighbourhood of samples.

Device-based US super-resolution The second harmonic image [TJ04] contains less noise and blur than the first harmonic image. Furthermore, the lateral resolution is increased, as the harmonic pulse is auto-focused because the higher harmonics are generated in the centre of the beam. Then, the image super-resolution is achieved by combining the first and second harmonic images. Both spatial and temporal deconvolutions [Lin04] are achieved by accounting for the transmit and receive processes, electrical transducer characteristics, and transmit focusing laws. Combining phase-contrast imaging, angular spectral decomposition, and a super-resolution reconstruction technique [CHH05], it is possible to recover the location and dimensions of objects smaller than the imaging wavelength. The reconstruction through generalised Tikhonov regularisation [LKO06] is evaluated as a function of transmit-receive bandwidth and a focal number of the transducer, by comparing the results with traditional B-mode imaging. The *Time-domain Optimized Nearfield Estimator* [VEW07] assumes an observation model based on the superposition of spatial responses; then, a maximum a-posteriori estimation finds the distribution and amplitude of hypothetical targets that match the observed data with minimal target energy. As a further improvement, the *Diffuse Time-domain Optimised Near-field Estimator* [EVW10] represents each hypothetical target in the system model as a diffuse region of targets rather than a single discrete target, thus inducing a better signal approximation. The cellular microscopy technique of multi-focal imaging [DGA⁺17] is applied to localise the unique position of the scatterer of the signal; three foci receive multiple overlapping curves, and a maximum likelihood estimation allows the identification of the source of the scatter.

Multi-frame US super-resolution The *Bilinear Deformable Block Matching* [BLM⁺08], which is a registration method that accounts for the complex and deformable motion of soft tissues, is applied to reconstruct the HR image by exploiting the shifting property of the Fourier transform and the aliasing relationship between the continuous Fourier transform of the HR image and the discrete Fourier transform of LR images [MBPK12]. The use of deep learning for motion estimation among different frames [ANO16] reduces the effect of noise and artefacts and reconstructs HR images from a sequence of LR images. The modelling of the spatial correlation of the speckle noise [CÖMS19] is applied to standard reconstruction methods, with tissue-mimicking phantom and co-registered multi-images.

2.2.3 Image approximation and sampling: background and related work

Point sampling is widely used in several Computer Graphics applications, such as point-based modelling [PKKG03] and rendering [SP04], image and geometric processing [PJH16, Mit87]. Point sampling is strictly related to image half-toning [LA01], which consists in simulating the full tone range of an image through a proper pattern of dots. This problem was raised with earlier mechanical printers, in order to reproduce the photographs in newspapers, using only one colour of ink for recreating a 256 grey levels image. The image is replicated by placing

more points in darker areas, according to local patterns, geometries, and colour intensities. The technical evolution of modern printers has achieved very good results in replicating continuous-tone images; for example, a modern laser printer can reach the resolution of 3K Dots per Inch. Point sampling has been applied also to image reconstruction [TV91], anti-aliasing [KH01], dithering [FH02], QR codes reconstruction [CCLM13], and artistic visualisation [OH95].

Important aspects of point sampling are *adaptation* to the input signal in order to guarantee that the sampling density is proportional to the image density or to the complexity of the signal in a given area; *feature preservation* without either over-smoothing or artefacts in the sampling or in the reconstructed signal; and *spectral properties* (e.g., blue-noise property) that allow us to achieve visually superior images, as the distribution of photoreceptors in a primate eye possesses the blue-noise characteristic [Yel83].

Signal sampling and approximation is applied in the biomedical context for registration [RHA07], features detection [SLW⁺20], and reconstruction [MU19] problems.

Approximation and sampling: related work

We briefly review previous work on signal sampling, according to six main classes: physics-based sampling, Gaussian mixture models and kernel-based sampling, stochastic sampling, kernel-based image sampling, tessellation-based sampling, and optimal transport.

Physics-based sampling is driven by physical equations, such as fluid dynamics, engineering, and electromagnetism. In [SGBW10], the particles (i.e., samples) are placed through a model inspired by electrostatics. The particles' attraction and repulsion are governed by the Coulomb laws, and the optimisation of the particles' position leads to an electrostatic equilibrium of the system. In [FHHA19], the particles' density is optimised by solving the Lagrangian formulation of the governing equations of compressible flow. In [FT16], the principles of natural selection acting on biological organisms are applied to the definition of a genetic algorithm for sampling, where the population (i.e., the samples) evolves with crossover and mutation events until an optimal solution is reached. In [Bos96], the interaction among particles is simulated through mechanical laws, including particles' interaction, and neighbouring effects. In [PS04], the mesh is optimised by solving a static force equilibrium in a truss structure, where the edges of the grid correspond to bars and the points correspond to the joints of the truss. The L_p -Gaussian kernels [ZZH19] reproduce the inter-particle energy to set up the sampling patterns, and the surface is reconstructed by exploiting the distribution of the particles. In [ZGW⁺13], the input anisotropic mesh is transformed into a high-dimension isotropic space; then, the mesh vertices are computed through the optimization of an energy functional. The sampling of images dominated by low frequencies [HSD13] exploits the link between Fourier analysis and spatial statistics, iteratively applying a force to the samples, that depends on the geometry of the samples,

without generating artefacts (e.g., aliasing).

Gaussian mixture models (GMMs) approximate an input signal as a mixture of probability distributions and have been successfully applied to different research problems such as segmentation [GRG06], denoising [TAF15], inverse problems [YSM12], inpainting [YSM11], registration [RGÇ⁺18]. GMMs are also relevant for clustering [BCG00], where the most challenging aspects are the variables' selection [MCMM09], initialisation [YLL12], and partitioning [SD07]. Finally, GMMs are ubiquitous in engineering [KS13], chemistry [DP20], biomedical [RB10], and signal processing [YKY13].

Kernel-based sampling In [She04], the samples are associated with kernel functions, whose linear combination with constant coefficients approximates the input signal. In [Fat11], the input signal is approximated in order to generate a point set with blue-noise properties. Each kernel function has a predefined support, and a statistical model is defined to allow solutions that further reduce the minimum of the energy functional. In [ZH16], Gaussian kernels with fixed support are applied to sample the input image, through an optimisation algorithm, whose variables are the samples' position. In [ÖAG10], the optimal sampling conditions are obtained by combining spectral analysis with kernel functions, and the samples are used to reconstruct a continuous surface with the desired smoothness. In [CP99], super-resolution images are reconstructed by applying a convolution operation and assuming a similarity among correlated neighbours. In [CGW⁺13], a sampling method with blue-noise properties is defined by considering both spatial and non-spatial properties and by modulating the samples' position with a domain-independent similarity. In [Han14], Gaussian kernels are used for approximating a continuous probability density function.

Stochastic sampling applies a probabilistic approach to generate a sampling where points are tightly packed, with a minimum distance constraint. Dart throwing [Coo86] places the samples sequentially; if the new sample does not satisfy the constraint, then it is rejected. This method can be applied to surface sampling [BWWM10] and image rendering [DW85]. In the hierarchical approach of the dart throwing algorithm [WCE07], the domain is subdivided into quad-tree regions, and the samples are placed only at active squares (i.e., where a sample is not already present). In [DH06], a very efficient algorithm for generating Poisson-disk distributions is achieved by representing the available neighbourhood for the insertion of a new sample through a data structure called scalloped region.

Kernel-based image sampling optimises the approximation accuracy of the input image, through the minimisation of an energy functional. In [PQW⁺08], the input image is compared with its sampling according to three metrics: luminance, contrast, and structure, integrated with

the tone similarity. In [FP04], a perceptually-based approach is proposed for progressive rendering, where samples are added through an iterative refinement, by computing the distance between the original image and its approximation with contrast maps corresponding to spatial frequency bands [PFFG98]. Constraints on the edge [Li06] and greyscale [AA92] preservation are also added to improve the quality of the image sampling.

Tessellation-based sampling methods apply a centroidal Voronoi tessellation, whose vertices provide a sampling of the input domain. Lloyd’s method [Llo82] computes the centres of the tessellation through an iterative algorithm, where the vertices are updated by computing the mass centroid of the Voronoi regions. In [DHVOS00], the Lloyd’s algorithm is specialised in half-toning applications. In [BSD09], a variant of Lloyd’s method is proposed by imposing that each point has the same capacity, which is defined as the area of the related Voronoi region, weighted with a density function. This method enhances the blue-noise characteristics and the density function adaptation of the sampling. In [Sec02], a variant of Lloyd’s method is defined by considering weighted centres. In [DO07], the Voronoi tessellation is built according to a density function defined on the input domain. In [CYC⁺12], a variational approach is defined by combining the centroidal Voronoi tessellation with the capacity-constrained one. In [SG95], bubbles are placed on the domain and are governed by inter-bubble forces; when a stable configuration is reached through a dynamic simulation, the tessellation is formed by connecting the centres of adjacent bubbles. In [CZC⁺18], surface sampling and reconstruction are computed through iterative centroidal Voronoi tessellation, based on a local approximation of the surface with a best-fitting plane. The generation of blue-noise sampling through Wang tessellation [KCODL06] is achieved with a recursive approach, which adaptively splits the tiles and relaxes the tessellation to match the point set.

Optimal transport [KMM⁺03] is applied to generate high dimensional sampling for the computation of the Monte Carlo integration of a generic function. In [DGBOD12], the capacity-constrained Voronoi tessellation is formulated as a continuous minimisation problem based on optimal transport and is applied to generate a blue-noise sampling without local artefacts. The optimal transport problem is progressively solved on a sequence of discrete measures [Mér11] that converge to the optimal solution. In [PBC⁺20], the sliced optimal transport projects and solves the problem onto repeated 1D dimensions; in particular, the distance to be minimised is defined as the integral over the slice directions between the projections of the input points on the selected direction and the orthogonal projection of the density function. In [QCHC17], a multi-class sampling is computed as a constrained barycentre of probability measures.

2.3 AI, HPC, and ultrasound imaging standards

We introduce artificial intelligence (Sect. 2.3.1), high-performance computing (Sect. 2.3.2), and imaging standards (Sect. 2.3.3) background for the management and processing of ultrasound data.

2.3.1 Artificial intelligence

Nowadays, the use and applicability of artificial intelligence for signal processing, in particular images and videos, is widespread through the availability of large data sets and the development of hardware and parallel computing techniques. The development of artificial intelligence methods that are efficient for the processing, approximation, and classification of images becomes relevant. Artificial intelligence methods take advantage of image redundancy, recognise complex patterns, and provide quantitative assessment for images (e.g., classification), showing high potential to assist the user in the acquisition of accurate and reproducible results. Machine learning (ML) is a branch of artificial intelligence that deals with large data sets to create systems that learn or improve their performance. Through the optimisation of a network of variables, ML methods allow us to learn a proper transformation from an input set to an output set. Deep learning is a subset of ML that processes and automatically learn mid-level and high-level abstract features from large data sets of images through multi-layered neural networks.

The area of AI in health and healthcare has seen rapid developments due to the increasing availability of health data, combined with unprecedented advances in AI and computation capabilities. AI systems can support humans allowing them to optimise the diagnostic examination workflow and improve the analysis of diagnostic images. The most recent algorithmic progress improves the quality of the acquired data and reduce the scan time; moreover, they can speed up the diagnosis and increase the diagnostic accuracy, by supporting the physician in the analysis of the images (e.g., ultrasound and magnetic resonance) and in the analysis of laboratory or histological data. Through quantitative information extracted by AI tools, radiologists might spot problems that are not immediately obvious to the human eye, especially in the case of advanced imaging techniques, such as functional imaging. AI, ML, and DL allow medical specialists to integrate imaging data with information from electronic medical records (e.g., longitudinal patient data, population health platforms) to provide a rich source of new insights for medical research, such as the identification of new protocols for imaging. Combining acquired data with predictive analysis, ML and DL methods provide significant benefits also supporting the management and prioritisation of medical analysis, improving intervention planning and preventing unnecessary re-admissions and further testing. From the technological perspective, all these aspects imply the need to expand their intra-operative capabilities, create operative equipment compatible with different imaging methodologies or with the procedural environment, and develop devices/accessories for navigation and interaction.

In this context, deep learning is widespread in the analysis of US images, for lesion/nodule classification, organ segmentation, and mass detection. The application of deep learning and task-specific clinical knowledge [LGL⁺19] allows us to automatically detect and classify thyroid nodules, with different types of nodule diagnosis and achieves better performances than experienced radiologists; in [LBK⁺18], the presence of metastatic lymph nodes is evaluated through DL techniques as a prognostic indicator for thyroid carcinomas. In the breast anatomical district, several works apply learning models for the classification of tumours; transfer learning and colour conversion [BGOF⁺19] are applied to classify tumours, comparing the results with the assessment of experienced radiologists. The segmentation of ROIs is also crucial to help the physicians and processing algorithms to evaluate and classify tumours; Sobel operators are applied to obtain a gradient magnitude image [GLR⁺16] to classify breast tumours into three major types: cyst/mass, fatty tissue, and fibro-glandular tissue. In the obstetric anatomical district, several clinical assessments are performed through AI techniques, such as respiratory morbidity [BAPMCG⁺19], identification and classification of fetal standard planes [YTN⁺18], and prediction of follicles' maturity during the reproductive cycle [LFL⁺20]. Other applications of deep learning in US concern texture analysis for the prediction of hypertension in pregnancy [GBL⁺21], quantitative analysis for evaluating COVID-19 pneumonia [CHY⁺21] and spontaneous preterm birth [BPMJ⁺18], co-registration with 3D MRI and CT [WHA⁺21], real-time guidance for screw insertion for scaphoid bone fixation after a fracture [ASR⁺15]. The application of deep learning methodologies in ultrasound processing requires attention to the design of the training data set, metrics, and learning architecture, to prevent the generation of patterns/artefacts. For a further review of deep learning methods for ultrasound processing, we refer the readers to [LWY⁺19, vSCE20].

Deep learning background Deep learning is a class of machine learning methods with multiple levels of representation, and it is composed of several non-linear layers that, starting with the raw input, transform the representation from the previous level into a representation at a higher level, with a reduction of the number of parameters that are applied to represent the image. With the account of large data sets and the composition of a large set of layers, deep learning methods attain very complex features and inferences. In this class of methods, one of the main distinctions is between image-to-label and image-to-image models: the goal of image-to-label models is to classify the training data according to two or more classes, for example, objects family or biomedical pathologies; the goal of image-to-image models is to start from an image and reconstruct another image as output with certain properties, as denoising or super-resolution problems. Deep learning methods are also classified between supervised and unsupervised, according to the prior knowledge of the output label/image.

Among deep learning methods, the convolutional operator is widespread for its property of local description and interpretation of images; its application leads to a sub-class of deep learning models named convolutional neural networks (CNNs). The convolution operator captures important features inside the images, such as contrast pixels and edges that are more representative of

the objects in the image, resulting in the reduction of memory use and computational cost. Also, the convolution operator allows to share the parameters (i.e., the variables of the model) among different patches of the images; this property allows the network to account for the information redundancy inside each image.

Each layer of a convolutional network generally performs three steps (Fig. 2.4). In the first step, the layer applies multiple convolutions through a set of operators (i.e., kernel filters). Next, the pooling function applies further changes to the output of the layer, according to the statistical values of nearby outputs. For example, *max pooling* modifies an output value according to the maximum value of its rectangular neighbourhood region, while *mean pooling* considers the average value of the neighbourhood region. The pooling operator generalises features extracted by convolutional filters and helps the network to recognise the features independent of their location in the image, thus emphasising the convolutional aspect of CNNs where neighbourhood values have an impact on any given node. Finally, activation functions introduce a non-linear element into the processing of the data of the network that allows each layer to operate on the input data and feed or not the output information to the next layer. The role of an activation function is to produce a mapping from an input real number to a real number within a specific range to determine whether or not the information within the node is useful. The values produced by the activation function depend on the type of activation function which is used, e.g., sigmoid, rectified linear unit, softmax.

Each kernel filter is associated with a set of variables (i.e., weights) that are optimised until the network is able to approximate the desired output with a proper accuracy. The number and dimension of each kernel depend on the architecture/purpose of the network, and these are two of the main hyper-parameters to be optimised for the improvement of the network's accuracy. Additional relevant hyper-parameters are the number and connections of the layers (i.e., the depth of the network) and the accuracy of the optimisation method.

Given a network with L layers, a set of kernel filters F each of size $m \times n$, usually selected as odd numbers (e.g., 3×3), we apply the convolution at layer l of the filter f to the input $X(l)$ as

$$X_{i,j}(l+1) = \sum_{h=1}^m \sum_{k=1}^n f_{h-\lceil m/2 \rceil, k-\lceil n/2 \rceil} \cdot X_{i,j}(l).$$

Some relevant parameters are the padding, which allows the convolution operator to process boundary values, and the stride, which allows the operator to process all the data or a selection of it. In addition, the bias adds a variable as a linear component of the convolution filter to improve the approximation of linear transformations between consecutive layers. Finally, the dropout regularisation is a computationally cheap way to regularise a deep neural network by probabilistically removing inputs to a layer and generating more robust networks.

We introduce forward propagation, where the data is provided to the first layer and advances the network through the designed layers. The starting values of the variables are initialised through a

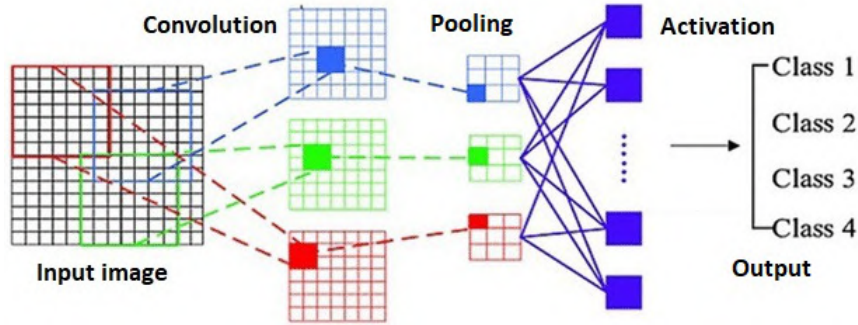


Figure 2.4: Example of Convolutional Neural Network.

proper method (e.g., normal distribution), and then optimised through an optimisation algorithm (e.g., Adam [Zha18]). After the computation of the predicted output \hat{y} of the network, this value is compared with the expected output y through a loss function, as mean squared error (Eq. (2.3)), mean averaged error, or logarithmic-based error. With n input data samples, the loss function is defined as

$$\mathcal{L} = \frac{1}{n} \sum_{i=1}^n \|y_i - \hat{y}_i\|^2. \quad (2.3)$$

Backpropagation is one method of training a network by minimising the loss function, which represents the accuracy of the network to predict the desired output. To minimise the loss, a gradient descent optimisation algorithm [Rud16] is applied and the errors are back-propagated from the last up to the first layer of the network. Gradient descent computes each weight as the negative of the learning rate multiplied by the partial derivative of the loss function with respect to the weights. Backpropagation reduces the number of calculations required to compute the gradients compared to forward propagation and reduces the approximation error of the network.

Among image-to-image deep learning methods, several architectures are defined by the number of layers, their connections, and the setting of the hyperparameters. Autoencoders (Fig.4) are widespread for their efficiency and are composed of two main steps: the encoding phase projects the information extracted by convolution operators into a smaller representation, composed of a reduced number of elements in a higher-dimensional space; the decoding phase projects back to a dimensional space and resolution as the expected output data, which may or may not be the same as the input. The optimiser learns the reduced number of variables that codify the representation of the image from the encoded data to the decoded data. Furthermore, connections among layers of the decoding and encoding phase allow the network to increase the accuracy of the approximation.

Generative adversarial networks (GANs) efficiently represent the training data, making it more effective at generating predictions similar to the expected target of the data set (Fig. 2.6). A GAN is composed of two concurrent networks: a discriminator D and a generator G . The generator,

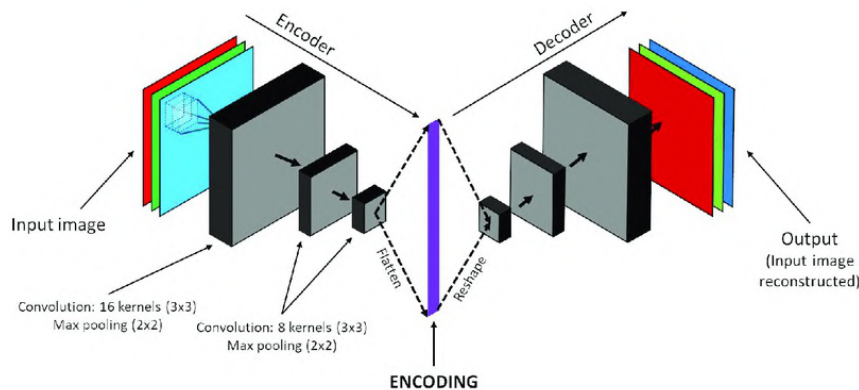


Figure 2.5: Example of an auto-encoder network. Image courtesy of [JTJT20].

for example, an autoencoder network, generates a prediction from an input image; the optimiser trains the network to generate a predicted output that resembles the expected output (i.e., the target of the data set), according to a defined loss. The discriminator is a network that is trained to classify target images and generated images; the weights of the network are optimised to recognise the first ones from the second ones. The communication between the two networks is bidirectional and concurrent: the generator provides its output to the discriminator, which classifies it as a target or generate. At the same time, the discriminator provides its feedback to the generator as a quantification of the number of fake images that have been correctly classified as the output of the generator itself. Unlike standard CNNs, where the measure of the accuracy of the prediction is performed through a static indicator (e.g., the mean squared error), GANs analyse the accuracy of the prediction with respect to the target through an additional network, which increases the accuracy of the prediction with respect to the expected target. A complete survey on deep learning methodologies for image processing is presented in [PC20].

Deep learning methods for image-to-image regression The *VGG19* [SZ14] introduces *Convolutional Neural Networks* (CNN) pushing the depth to 16–19 weight layers, using small convolution filters of 3×3 size, with an application to large scale images classification. The *Pix2Pix* [IZZE17] method is a *Generative Adversarial Network* (GAN), where the generator is a U-net [RFB15], the discriminator is an encoding network [KSH17], and the loss function is based on the binary cross-entropy. The *deep convolutional generative adversarial network* [RMC15] applies unsupervised learning for image classification and the generation of natural images, exploiting batch normalisation, rectified linear unit activations, and removing fully connected hidden layers. CycleGan [ZPIE17] learns to translate between domains without paired input-output examples. The addition of a perceptual loss as an optimisation target of the generator [GEB16] allows the network to improve structure preservation. The introduction of a spectral normalisation as weight normalisation technique [MKKY18] allows to stabilise the training of the discriminator in a GAN. For a further discussion of image-to-image deep learning methods, we refer the

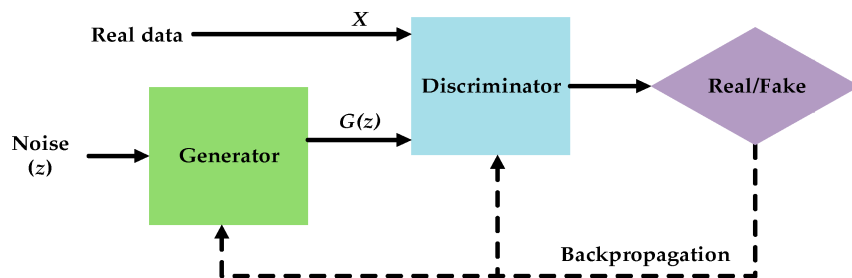


Figure 2.6: Example of Generative Adversarial Network. Image courtesy of [FFC⁺20].

reader to [KK19].

2.3.2 High-performance computing

Nowadays, technological evolution allows us large data acquisition that is elaborated by complex multidisciplinary applications. Parallel computing enables us to solve big data problems exceeding the limits of memory and reducing the computation time through the synchronised usage of more processes. The method and/or the data are divided into multiple portions that are allocated to one or more processes and solved concurrently. In this context, the application of high-performance computing (HPC) methodologies, the accounting for modern HPC resources (e.g., computing clusters, cloud computing), and the development of proper parallel implementations of processing algorithms become relevant to accelerate intensive computation for solving problems in numerous data science applications.

Modern HPC resources are typically highly heterogeneous systems, composed of multiple CPUs and accelerators; the latest generation CPUs are hierarchical, accounting of multicore processors and supported by powerful specialised accelerators, such as graphics processing units (GPUs). These resources offer significant computational power in the order of TFLOPS, which is achieved by exploiting such huge parallel architectures. The development of efficient software takes into account efficient data management; the computational performance is dominated by communications among processes and memory traffic, rather than floating-point operations of the individual process; moreover, the different kinds of processors and accelerators pose new and major issues in terms of memory management and data access.

Over the past decade, HPC has been increasingly introduced into image processing and biomedical applications to solve computational and big data challenges. In this context, HPC is widespread for the training of learning models in US processing; for example, for the localisation of common carotid artery transverse section through RCNN [JGB⁺20], automatic segmentation of the carotid artery and internal jugular veins [GVV⁺20], fetal standard planes recognition [PLLZ21], and segmentation and classification of anatomical structures [PBA⁺19]. HPC

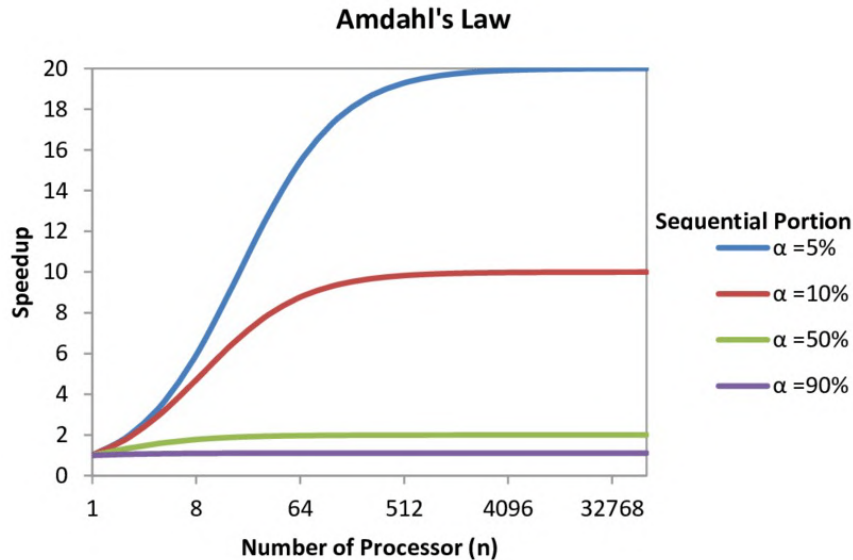


Figure 2.7: Amdahl’s Law showing the speed-up (y -axis) with respect to the number of processes (x -axis) for different fractions of sequential code.

and cloud computing also poses new challenges in terms of medical data privacy and security. For a deeper analysis of HPC for biomedical data, we refer the reader to the DeepHealth project [CGB19].

High-performance computing background The main reasons for using a parallel environment are the access to larger memory and higher computational power. Parallel computing can be divided into two main categories: data parallelism and model parallelism. In the first case, all the processes execute the same algorithm on different data, while in the second case all the processes access the same data while executing different parts of the algorithm. The choice between the two approaches depends on the complexity of the data rather than the independence of the algorithm’s subroutines. In both cases, communication and balancing among processes and data distribution become relevant for optimising the computational cost without facing bottleneck issues. Unfortunately, the memory and/or the overall computing power can not be calculated as the simple sum of the individual components, due to the complexities arising from the communication between the processes and the sharing or duplication of the necessary data.

To measure the efficiency of a parallel implementation, we introduce some quantitative measures that account for the computational performance with respect to single-process execution. Given p processes, and defining T_1 as the execution time of an algorithm on a single processor and T_p as the execution time on p processors, we define the *speedup* as $S_p = T_1/T_p$. In practical cases, $S_p \leq p$, since the execution time on p processes is higher than the execution time on

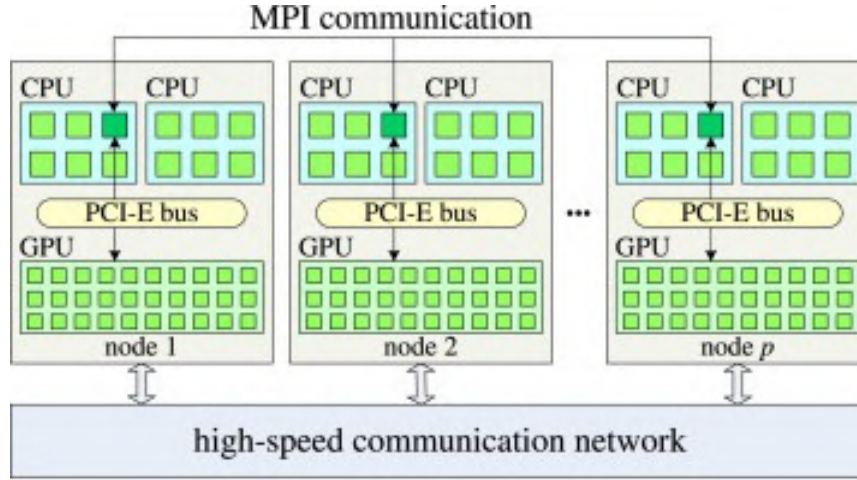


Figure 2.8: Example of HPC architecture with CPUs, GPUs, and inter-nodes communication. Image courtesy of [WLL16].

one process divided by the number of processes, i.e., $T_p \leq T_1/p$. To quantify the *efficiency* of our parallelisation, we introduce a quantitative measure $E_p = S_p/p$ that accounts the distance of our parallelisation from an ideal distribution of the algorithm among the processes; in fact, $E_p \leq 1$. There exist different reasons for the loss of efficiency in parallelisation: the use of different processes includes the addition of communication among them and synchronisation in the data communication that leads to barriers in the method and consequently to an increase in the execution time. Then, the load unbalance among processes implies that they do not have exactly the same amount of work, leading to idle time for some processes. Finally, the algorithm has subroutines that are inherently sequential and their execution times can not be reduced independently of the number of processes adopted. The addition of these overheads implies a loss of efficiency in the parallel code.

This result is summarised by Amdahl's Law (Fig. 2.7). Given F_s be the sequential fraction of a code, and F_p the parallel fraction with p processes, with $F_p + F_s = 1$, the global execution time T_p is given by the execution time of the sequential part $F_s T_1$ plus the execution time of the parallel part $F_p(T_1/p)$, where T_1 is the execution time of one process; as result: $T_p = T_1(F_s + F_p/p)$. In addition, the parallelisation introduces a communication overhead among processes, that further reduces the speed-up. Amdahl's Law is then refined as

$$T_p = T_1(F_s + F_p/P) + T_c,$$

where T_c is a fixed communication time and establishes that the number of processes should not grow over the ratio of scalar execution time and communication overhead, since the *efficiency* tends to decrease as the number of processes increases.

HPC resources for deep learning HPC resources are usually composed of multiple nodes, each one composed of a set of multi-core CPUs and a set of GPUs. Each node shares the same private memory, while the passage of data between different nodes is managed through inter-nodes communication methods (Fig. 2.8). The parallelism inside a single node is more efficient when the number of processes is not greater than the number of cores since the communication among processes is faster but the quantity of memory is limited. On the contrary, multi-node parallelisation allows the HPC model to account for an increased quantity of memory and to apply a higher number of processes.

In this context, we can exploit HPC for deep learning as

- *Data parallelism* by splitting the images among different processes;
- *Model parallelism* by splitting the algebraic operations among different processes.

The *data parallelism* involves the distribution of the images among the nodes of the computing resource. This allows us to increase the batch size, i.e., the number of images loaded and processed simultaneously for each process. The *model parallelism* involves the parallelisation of the operations performed by the layers of the network. In particular, the convolution operation between the input data \mathbf{X} and the convolution filter \mathbf{K} is equivalent to multiplying the input data by a Toeplitz matrix. Given \mathbf{X} as a $n \times n$ matrix, and \mathbf{K} as a $m \times m$ matrix, with $n = 3$ and $m = 2$, we unroll \mathbf{k} into a sparse matrix of size $(n - m + 1)^2 \times n^2$, and \mathbf{X} into a $n^2 \times 1$ vector. We can compute the sparse matrix-vector multiplication as

$$\begin{pmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} & 0 & \mathbf{K}_{21} & \mathbf{K}_{22} & 0 & 0 & 0 & 0 \\ 0 & \mathbf{K}_{11} & \mathbf{K}_{12} & 0 & \mathbf{K}_{21} & \mathbf{K}_{22} & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{K}_{11} & \mathbf{K}_{12} & 0 & \mathbf{K}_{21} & \mathbf{K}_{22} & 0 \\ 0 & 0 & 0 & 0 & \mathbf{K}_{11} & \mathbf{K}_{12} & 0 & \mathbf{K}_{21} & \mathbf{K}_{22} \end{pmatrix} \begin{pmatrix} \mathbf{X}_{11} \\ \mathbf{X}_{12} \\ \mathbf{X}_{13} \\ \mathbf{X}_{21} \\ \mathbf{X}_{22} \\ \mathbf{X}_{23} \\ \mathbf{X}_{31} \\ \mathbf{X}_{32} \\ \mathbf{X}_{33} \end{pmatrix} = \begin{pmatrix} \mathbf{K}_{11}\mathbf{X}_{11} + \mathbf{K}_{12}\mathbf{X}_{12} + \mathbf{K}_{21}\mathbf{X}_{21} + \mathbf{K}_{22}\mathbf{X}_{22} \\ \mathbf{K}_{11}\mathbf{X}_{12} + \mathbf{K}_{12}\mathbf{X}_{13} + \mathbf{K}_{21}\mathbf{X}_{22} + \mathbf{K}_{22}\mathbf{X}_{23} \\ \mathbf{K}_{11}\mathbf{X}_{21} + \mathbf{K}_{12}\mathbf{X}_{22} + \mathbf{K}_{21}\mathbf{X}_{31} + \mathbf{K}_{22}\mathbf{X}_{32} \\ \mathbf{K}_{11}\mathbf{X}_{22} + \mathbf{K}_{12}\mathbf{X}_{23} + \mathbf{K}_{21}\mathbf{X}_{32} + \mathbf{K}_{22}\mathbf{X}_{33} \end{pmatrix}$$

and convert the result into a $n - m + 1$ square matrix

$$\begin{pmatrix} \mathbf{K}_{11}\mathbf{X}_{11} + \mathbf{K}_{12}\mathbf{X}_{12} + \mathbf{K}_{21}\mathbf{X}_{21} + \mathbf{K}_{22}\mathbf{X}_{22} & \mathbf{K}_{11}\mathbf{X}_{12} + \mathbf{K}_{12}\mathbf{X}_{13} + \mathbf{K}_{21}\mathbf{X}_{22} + \mathbf{K}_{22}\mathbf{X}_{23} \\ \mathbf{K}_{11}\mathbf{X}_{21} + \mathbf{K}_{12}\mathbf{X}_{22} + \mathbf{K}_{21}\mathbf{X}_{31} + \mathbf{K}_{22}\mathbf{X}_{32} & \mathbf{K}_{11}\mathbf{X}_{22} + \mathbf{K}_{12}\mathbf{X}_{23} + \mathbf{K}_{21}\mathbf{X}_{32} + \mathbf{K}_{22}\mathbf{X}_{33} \end{pmatrix}$$

This operation has lower complexity than a fully general matrix-matrix multiplication and is parallel among the processes inside each node.

2.3.3 Imaging standards

Nowadays, with the development of different imaging techniques (e.g., US, MR), numerical technologies are dominant in diagnostic imaging. Thus, the management of medical images is

inherently based on information technologies, making computers at the core of generation, memorisation, transmission, and access to those images. In this context, it is necessary to determine a universal standard system for the transmission of biomedical images. All the companies supplying equipment for the production of images apply as communication protocol the digital imaging and communications in medicine (DICOM) standard to guarantee a high interconnection and interoperability capability. The DICOM standard is composed of a multi-part document made of 18 modules (Fig. 2.9). *Part 3 Information Object* definitions specify the information objects to be exchanged and the definition of the semantics of each data element. This part is very long due to the various existing imaging modalities, that require many technical parameters. *Part 4 Service Class* specifications describe the services for exchanging information, like the images themselves or other information useful to manage the images. *Part 16 Content Mapping Resource* defines the terminology, specifically how existing terminological resources can be used in DICOM, and how to reuse grouped content items in DICOM Structured Reporting documents. *Part 5 Data Structure and Encoding* describes how to organise the information objects specified in Part 3 into a linear bitstream, in order to be sent over a network connection or stored in a file. This comprehends all aspects related to image compression. Finally, *Part 2 Conformance* specifies how to claim conformance to the DICOM standard for a particular product, meaning how to write the document called conformance statement in detail. This standard not only defines a transmission protocol for images but also regulates the memorisation of those images. Together with the image, this standard permits the achievement of different other pieces of information that identify the patient, the exam parameters, and the image itself. Indeed the information of the DICOM element is composed by:

- a tag that identifies the element and the group to which the packet belongs to. Where the group is the DICOM entity associated with the data and the element is the type of attribute contained in the packet. As an example, group 0100 identify the patient entity, and in this entity, element 0010 identifies the surname;
- a type indicator that identifies the format of the information memorisation;
- the byte length of the transmitted information;
- the real principal information in the format indicated by the previous fields.

Besides the public tag, which is present in all the data saved through the standard, there are also private tags, that are inserted by the specific company, and contain information that depends on the specific implementation. However, such information can be fundamental for the processing and correct visualisation of the image. An in-depth analysis of DICOM standard [Pia08] discusses several properties, such as communication protocols, storage and archiving, encoding and compression, anonymisation, encryption, and disaster-proof design.

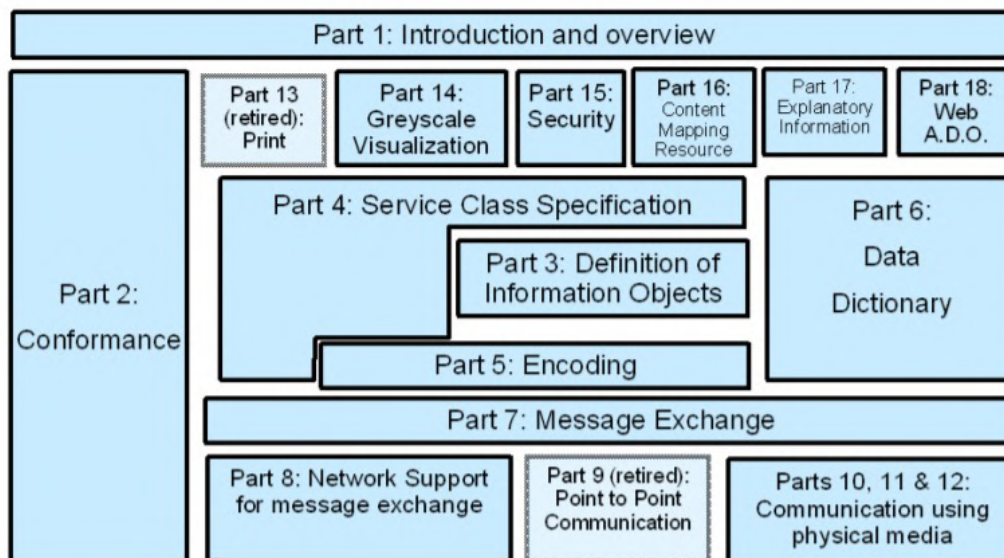


Figure 2.9: DICOM standard multi-document organization [Gib08].

Data security The availability of large amounts of data allows us to apply artificial intelligence techniques for signal processing. While a lot of synthetic and natural data are available, the access to medical data is limited, both for patients' privacy reasons and for the exclusivity of the data linked to hospitals and medical companies. Large biomedical data sets provide huge opportunities to improve the health sector; data analytics allow hospitals to monitor diseases and outbreaks, predict patients' behaviours, and provide risk stratification for individual patients.

In this context, the detail and diversity of information collected in the healthcare system and biomedical research are largely increasing, with clinical and administrative health data being complemented with radiomics data. The processing of large amounts of data in the biomedical field must take into account the privacy and protection of information. In fact, the data sets contain metadata regarding the patient's personal and clinical information, such as patients' identification, their psychological or medical conditions, and medical services provided. Data privacy in healthcare is critical since keeping patients' information secure and confidential helps in building trust between patients themselves and the whole healthcare system, including physicians and hospital services. Also, the evolution of technologies for cloud storage and the online processing of data support the development of blockchain cryptography [SS21] and communication protocols [AHRH⁺17]. Some advances in biomedical privacy problems and perspectives are discussed in [MEO13] and [Mit19].

2.4 Experimental set-up

Data set of ultrasound signals

We refer to *2D images* as $m \times n$ matrices, *2D videos* as $m \times n \times t$ tensors, where each frame has a $m \times n$ resolution, and t frames are acquired. We refer to *3D (volumetric) images* as $m \times n \times p$ tensors, where p planes are acquired, and each plane has a $m \times n$ resolution. Finally, 3D meshes can be extracted from 3D images; in this case, the signal is represented as a set of vertices \mathcal{V} and an adjacency matrix \mathcal{K} , e.g., a set of triangles.

Biomedical data For our tests, Esaote made available a large data set of images, videos, and volumetric images, belonging to different anatomical districts. This allows us to apply the learning techniques and specialise the developed methods on the type of data and anatomical district. In particular, our data set contains more than 10K ultrasound images at different resolutions and is acquired from different (e.g., muscle-skeletal, obstetric, abdominal) anatomical districts. Also, we consider 2D videos (e.g., 90 cardiac district videos, $600 \times 300 \times 170$ frames exemplary resolution, and 3D images (e.g., 80 obstetric district 3D images, $600 \times 500 \times 128$ exemplary resolution).

Synthetic data In some contexts, synthetic signals are required for testing specific methods. Synthetic signals shall respect the properties of US signals, that are affected by speckle noise. As synthetic data, we consider the *SIFI data set* [Web97], which is composed of 44 ground truth images of different sizes, organised in different classes (e.g., humans, landscapes). We apply artificial speckle noise with different levels of intensity: given a noisy image $Y = X + NX$, where X is the normalised ground truth image, we define the artificial multiplicative noise $N(x) = \sqrt{12}\sigma u$, where $u \sim \mathcal{U}(-0.5, 0.5)$, \mathcal{U} is a uniform distribution, σ is the noise intensity, and x is a pixel of the image. We also consider volumetric images (e.g., phantom cyst) and 2D videos (e.g., man talking): on volumetric images, noise is applied slice by slice, while in 2D videos, noise is applied frame by frame. Finally, we test some of our methods with different types of noise: Gaussian, Salt & Pepper, Exponential, and Poisson.

Hardware and software

HPC Cluster We exploited the large computational performance of the CINECA cluster to apply our methods to a parallel environment. The CINECA Marconi100 cluster occupies the 21st position in the *top500* list [urla]. The cluster uses 980 nodes, each with IBM Power9 AC922 at 3.1GHz 32 cores and 4x NVIDIA Volta V100 GPUs per node, with NVlink 2.0 16GB. We have

access to 40K hours on Marconi V100 through two ISCRA-C projects: *US-ML-SR* and *IMG-DEN*. We have performed the training on the Marconi V100 cluster, exploiting the HPC resources on both CPUs and GPUs through a parallel implementation of our deep learning framework on multiple nodes.

Workstation We perform some of our tests on a standard workstation with 2 Intel i9-9900KF CPUs (3.60GHz), 1 Nvidia GeForce RTX 2070 GPU, and 32 GB RAM.

Software We implement our training model in Python, through the scientific library Tensorflow 2.7.0. The low-rank denoising method is implemented in Matlab 2020a.

PART II

**Real-time US denoising:
contributions and novelties**

Chapter 3

Real-time denoising of 2D US images

Ultrasound imaging uses high-frequency sound waves to visualise soft tissues, such as internal organs, and support medical diagnosis for muscle-skeletal, cardiac, and obstetrical diseases, due to the efficiency and non-invasiveness of the acquisition methodology. Ultrasonic sound waves are reflected off from different layers of body tissues. The main issues of the ultrasound techniques are a significant loss of information during the reconstruction of the signal, the dependency of the signal from the direction of acquisition, and an underlying noise that corrupts the image and significantly affects the evaluation of the morphology of the anatomical district. In this context, the denoising of ultrasound images is relevant both for post-processing and visual evaluation by the physician.

Real-time framework Our main goal is the definition of a novel deep learning framework for the real-time denoising of ultrasound images (Fig. 3.1). After the design of a training data set, composed of raw images and the corresponding denoised images, we train a neural network that replicates the denoising results. Then, real-time denoising is achieved through the prediction of the trained network. The proposed framework combines three elements: *low-rank denoising*, *deep learning*, and *high-performance computing* (HPC, for short).

We select WNNM-*Weighted Nuclear Norm Minimisation* [GZZF14] as the best denoising method, which is then specialised for ultrasound images as a “new” *tuned-WNNM* denoising, by tuning its parameters. The choice of WNNM is based on a qualitative and quantitative comparison of five denoising methods, i.e., WNNM, SAR-BM3D - *SAR Block-Matching 3D* [PPAV11], BM-CNN - *Block Matching Convolutional Neural Network* [AC17], NCSR - *Non-Locally Centralised Sparse Representation* method [DZSL12], PCA-BM3D *Principal Component Analysis Block Matching 3D* [DFKE09]) belonging to the spectral, low-rank, and deep learning classes (Sect. 2.2.1).

To achieve a *real-time denoising of ultrasound images*, we propose a deep learning framework

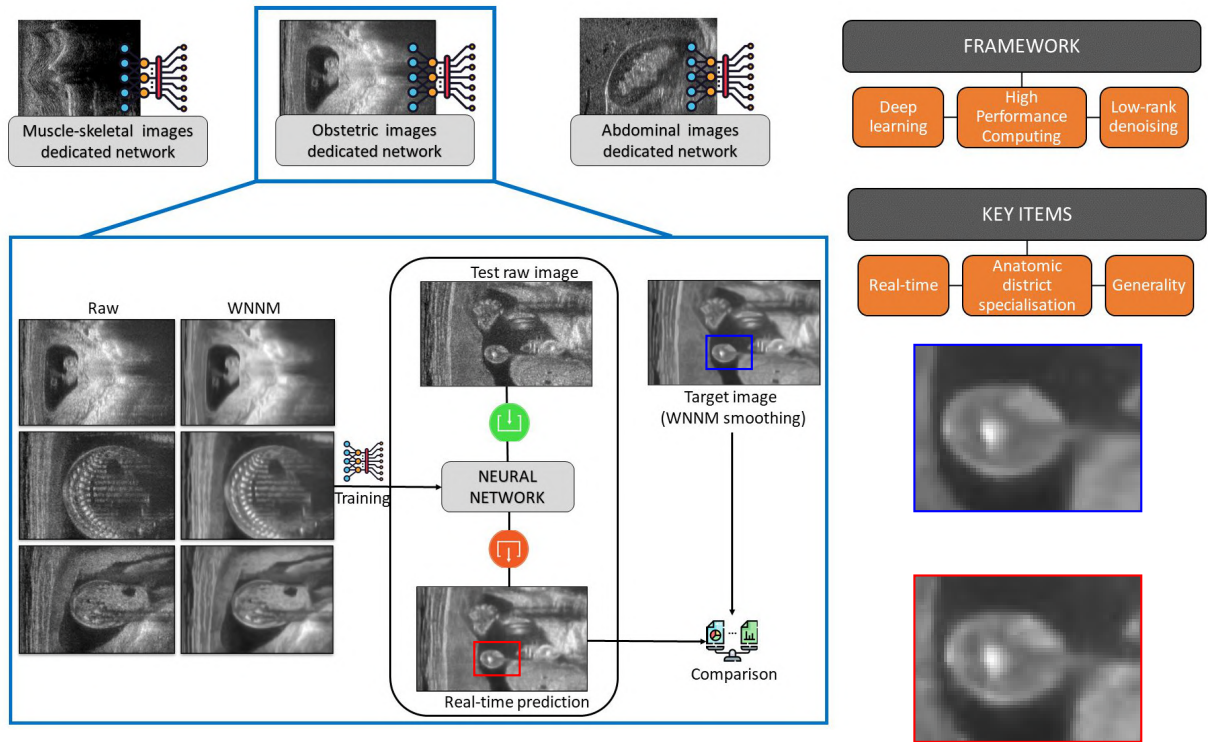


Figure 3.1: Pipeline of the proposed framework, with a magnification of prediction (red) and target (blue) images (right side). Our framework applies deep learning and HPC to learn and replicate the denoising results of the state-of-the-art low-rank method (i.e., tuned-WNNM), in real-time and with a specialisation to anatomic districts.

that is based on the learning of the tuned-WNNM and HPC tools (Sect. 3.1). The training is performed offline and can be further improved with new data, a-priori information on the input images or the anatomical district, and denoised images selected after experts' validation. Through our framework, the execution time of denoising only depends on the network prediction, which is achieved in real-time on standard medical hardware.

As the main contribution, the proposed denoising of ultrasound images runs in real-time and is general in terms of the input data, resolution of the input images (e.g., isotropic, anisotropic images), acquisition methodology, anatomical district, and speckle noise intensity. Our approach is also general in terms of the building blocks and parameters of the deep learning framework; in fact, we can select different denoising algorithms (e.g., WNNM, SARBM3D) and deep learning architectures (e.g., Pix2Pix, VGG19).

As experimental validation (Sect. 3.2), we perform a quantitative (e.g., PSNR, SSIM) and qualitative evaluation of the selected denoising methods on ultrasound images acquired from different anatomical (e.g., muscle-skeletal, obstetric, and abdominal) districts. Then, the results of the

deep learning and HPC frameworks are quantitatively and qualitatively analysed on a large collection of ultrasound images. The industrial requirement of real-time denoising is verified in terms of the execution time of the network prediction on GPU-based hardware installed in commercial ultrasound machines.

Novel denoising method As an original contribution to the denoising algorithm inside the framework, we propose a novel learning-based low-rank method for the denoising of 2D images, with an application to ultrasound images (Sect. 3.3). Our method improves the results of previous work and can be inserted into our framework in place of *tuned-WNNM* for the real-time denoise of ultrasound 2D images, demonstrating the generality of the framework to the denoise method.

In this context, our *goal* is the definition of a novel denoising method belonging to the low-rank class, based on the *Singular Value Decomposition* (SVD) and the learning and prediction of the optimal thresholds. Given a training data set of ground truth images (Fig. 3.2), we apply both speckle noise at several intensities and different types of noise (e.g., Gaussian, salt & pepper, Poisson, exponential) and extract 3D blocks through the block matching algorithm [DFKE06]. Then, we compute the optimal singular values through a proper optimisation applied to the SVD of each 3D block. We iterate this approach, where the input image of each iteration is the denoised image at the previous step.

The input and optimal singular values compose the training data set of the learning model; the singular values of the training data set are classified according to four parameters: noise type and intensity, number of iterations of the method, and clustering of the singular values. This classification allows us to design specific networks and improve the accuracy of the learning model. The learning phase optimises a matrix of weights, is applied to train the model and predicts the optimal thresholds of the SVD. For image denoising, we compute the 3D blocks and the SVD, the four parameters of each 3D block, and the thresholds through the network's prediction. Finally, the block-matching aggregation is applied to reconstruct the denoised image. As *main contributions*, we train our learning-based model to predict the optimal thresholds of the SVD. In contrast, learning-based methods [TFZ⁺20] predict the denoised image; low-rank methods [HNTL18] manually select the SVD thresholds; the *block-matching 3D* (BM3D) [DFKE06] applies a wavelet decomposition, with a manual selection of the thresholds. Our denoising method is independent of the type of noise and provides an upper bound to the accuracy of the denoising of 2D images through the SVD. Our approach can be generalised to different transformations of the noisy signal, e.g., spectral (wavelet, shearlet), factorisation (e.g., SVD, HOSVD), or to volumetric data through spherical harmonics and Laplacian eigenpairs. Trained networks are available at <https://github.com/cammarasana123/denoise>.

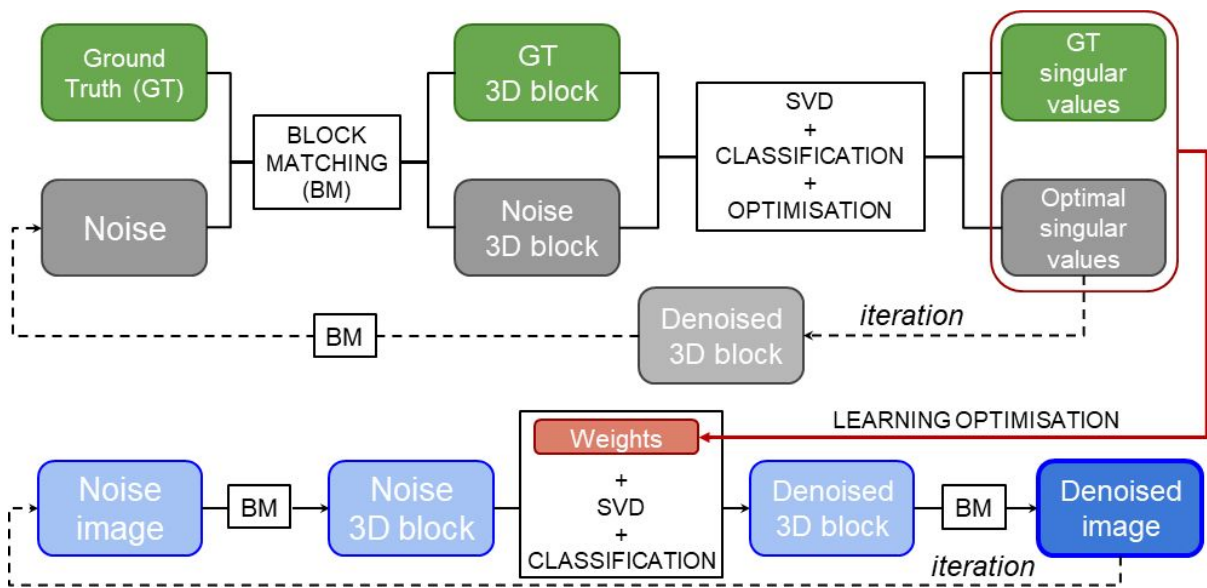


Figure 3.2: Learning-based for denoising: pipeline.

Related publications

Cammarasana, Simone, Paolo Nicolardi, and Giuseppe Patanè. "Real-time denoising of ultrasound images based on deep learning.", *Medical & Biological Engineering & Computing* (2022): 1-16. <https://doi.org/10.1007/s11517-022-02573-5> [Journal Paper]

Cammarasana, Simone, and Giuseppe Patane. "Learning-based low-rank denoising.", *Signal, Image and Video Processing* (2022): 1-7. <https://doi.org/10.1007/s11760-022-02258-4> [Journal Paper]

Cammarasana, Simone, Paolo Nicolardi, and Giuseppe Patanè , "Signal despeckling with learned regularisation.", *Subdivision, Geometric and Algebraic Methods, Isogeometric Analysis and Refinability in Italy (SMART)*, Rimini, September 2022. https://events.unibo.it/smart2022/program-and-abstracts/book_smart_2022.pdf/@@download/file/book_SMART_2022.pdf [Extended Abstract]

Cammarasana, Simone and Giuseppe Patanè, "Learning-based Low-Rank Denoising: An application to biomedical images.", *Computer Science Workshop (CSW)*, DIBRIS-University of Genova, June 2022. <https://docs-dibris.github.io/assets/theme/posters/Cammarasana.pdf> [Poster]

Cammarasana, Simone, Paolo Nicolardi, and Giuseppe Patanè. "A universal deep learning framework for real-time denoising of ultrasound images.", *arXiv preprint arXiv:2101.09122* (2021). <https://doi.org/10.48550/arXiv.2101.09122> [Technical Report]

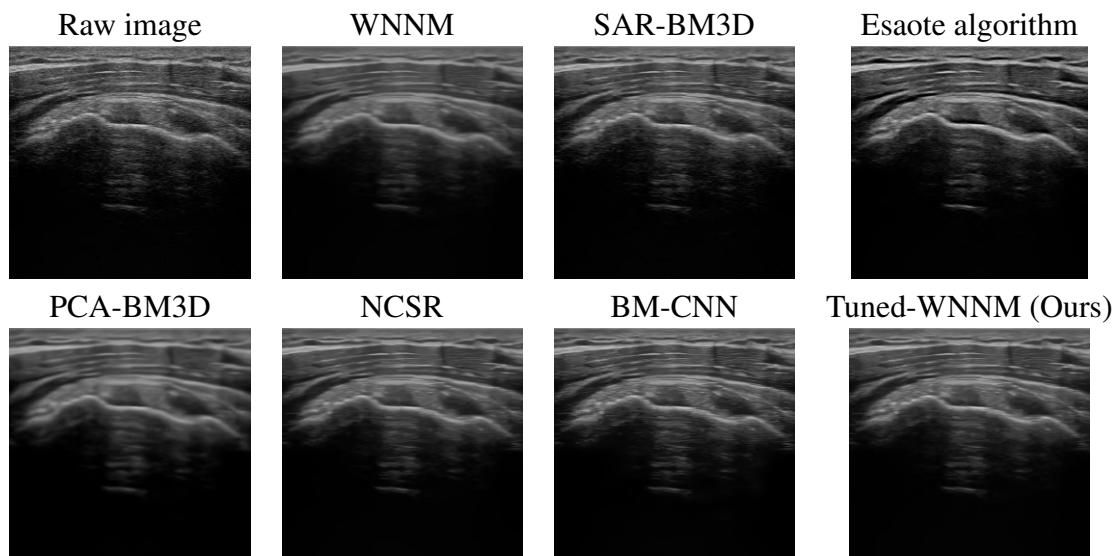


Figure 3.3: Raw images of a muscle-skeletal district and denoised images, visualised in a scan-converted format.

3.1 Real-time denoising: proposed framework

We propose a novel deep learning framework for the real-time denoising of ultrasound images. We discuss the motivation and contribution of our framework, the design in terms of deep-learning and HPC architectures, and the data set and metrics for the evaluation of the accuracy of the results. For a detailed discussion of the experimental results, we refer the reader to Sect. 3.2.

The main requirements of a denoising algorithm for ultrasound images are the magnitude of the removed noise, edge preservation, and real-time computation. The main denoising methods satisfy these requirements, except for the execution time, which does not satisfy the real-time need of ultrasound applications. To achieve real-time computation and to maintain the good results of the best denoising methods in terms of noise removal and edge preservation, we identify two strategies. After the identification of the denoising method that best fits our requirements for ultrasound images: (i) we develop a computationally optimised version of a selected denoising algorithm, exploiting HPC tools, CPUs and GPUs, and low-level programming languages; (ii) we design and implement a deep learning framework that learns to replicate the output of the selected denoising algorithm. In this context, we analyse state-of-the-art denoising methods to identify the one that best fits our requirements in terms of noise removal and preservation of anatomical structures and geometries.

Tuned-WNNM We compare five denoising state-of-the-art methods, which are either specific for speckle noise (e.g., SAR-BM3D) or independent of the type of noise (e.g., WNNM). Accord-

ing to the results in Sect. 3.2, WNNM has been selected as the best denoising method among five state-of-the-art methods belonging to the spectral, low-rank, and deep learning classes. To improve the quality of the denoised image, we propose a novel approach to the tuning of WNNM parameters, and we refer to this method as *tuned-WNNM*.

Given a pixel x , the *patch* P_x is the set of pixels in the neighbourhood of x ; each pixel of the image has a related patch. The *search window* is the set of patches selected for searching the closest ones to a reference patch, under a certain metric. The *stack*, or *3D block*, is the set of patches that are similar to a reference patch; these patches are stored in a 3D structure, and the redundancy of the stack is exploited to remove the noise. Within this setting, we identify and optimise the following parameters: the number of patches is no more limited by the step value (e.g., 1 patch every 2 or 3 pixels) and we assign a patch for each pixel; this parameter allows us to increase the number of processed patches, thus improving the data redundancy. The block-matching algorithm is now performed every iteration, instead of one every two iterations; the selection of the searching window and the size of the stack are now larger than in previous work. These parameters allow us to improve the measure of the similarity among 3D blocks and the accuracy of the denoising method. Furthermore, we specialise in the tuned-WNNM method for ultrasound images, by varying the denoising intensity through a parameter that affects the threshold of the singular values of the SVD.

The implementation of a computationally optimised, and potentially real-time, version of the tuned-WNNM is a very tough requirement; the main iterative cycle of the algorithm is not parallelisable, due to the dependency of the data among the iterations. Furthermore, the cubic computational cost for the evaluation of the SVD of each block is no further optimisable. The real-time requirement needs a specific hardware-based implementation, and any modification to the method requires a new implementation of the parallel algorithm. This approach needs dedicated and more expensive hardware, which is in contrast with the cheapness of ultrasound acquisition.

Proposed approach and contributions The proposed real-time denoising is based on the training of a neural network to learn and replicate the tuned-WNNM behaviour. In the first phase, the network is trained on a data set of ultrasound images of the same district. The data set for the training of the learning method is composed of a set of couples of ultrasound images: the input (i.e., the raw image) and the target (i.e., the image denoised with the tuned-WNNM filter). The ground truth is not available in ultrasound applications; for this reason, the target of the learning method is the output of the tuned-WNNM filter. Then, the trained network provides the denoised output through a real-time prediction of the test images. As the main contribution, the proposed deep learning framework is general in terms of the input data, i.e., type of noise (e.g., speckle, Gaussian noise), resolution (e.g., isotropic, anisotropic) of the input images, acquisition methodology, and the anatomical district. Our deep learning framework is also general in terms of building blocks and parameters: since different methods (e.g., NCSR, SAR-BM3D,

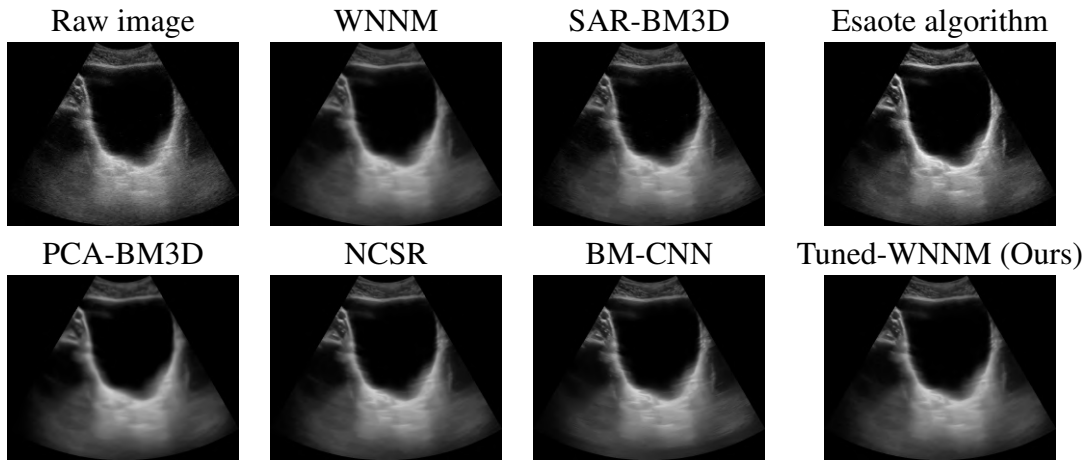


Figure 3.4: Raw data set of an abdominal district and denoised images, visualised in a scan-converted format.

custom methods) have good performances, the generality of our framework allows us to use a different denoising algorithm and exploit its different characteristics in terms of denoising and edge preservation. Alternative denoising methods can be used for different types of noise (e.g., speckle, Gaussian noise) and signals (e.g., 3D images, time-dependent ultrasound videos, see Chapter 4).

In our approach, we specialise the training phase to specific anatomical districts or types of noise. For instance, we train a specific network for each district, thus obtaining a more precise result when predicting the denoised image, as each network is specialised to the input anatomical features. We also improve the WNNM denoising in terms of real-time computation based on offline training. Our framework maximises the performance of the denoising method: in fact, our tuning improves the WNNM results with an increase in execution time; however, the real-time requirement is achieved by the network prediction, while the denoising is applied offline for the generation of the training data set. Our tuning does not affect the real-time execution of the proposed framework, which depends only on the execution time of the network prediction.

Deep learning architecture To evaluate the proposed framework, we analyse several networks and perform an image-to-image regression; among them (Sect. 2.3.1), we select Pix2Pix, which guarantees good results in terms of learning. We specialise Pix2Pix to ultrasound images, with two updates: (i) the introduction of a validation data set of the same district of the training data set, which forces the exit condition when the validation error increases, and (ii) the introduction of padding and masking pre-processing operations, which allow us to deal with images of different resolution, without an image resize that would imply a distortion artefact. In fact, the Pix2Pix architecture requires input images of constant size. Through padding, the images are resized according to the input required by the architecture; after that, a masking filter allows the

Table 3.1: Execution time computed as an average value of a set of Esaote images at 600×485 resolution.

| Method | WNNM | SAR-BM3D | BM-CNN | NCSR | PCA-BM3D | Esaote algorithm |
|--------------------|------|----------|--------|------|----------|------------------|
| Execution time [s] | 215 | 55 | 356 | 380 | 95 | real-time |

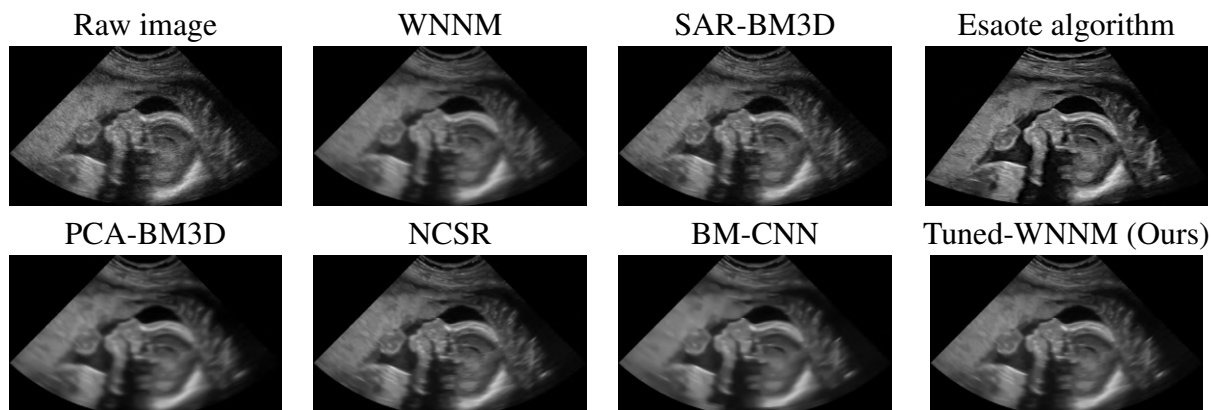


Figure 3.5: Raw data set of an obstetric district and denoised images, visualised in a scan-converted format.

network to process only the numerical values of the image and not the values added by padding. A comparison between Pix2Pix and *Matlab* CNN is discussed in Sect. 3.2.2.

HPC framework for learning The experimental tests on the CINECA Cluster are supported by the ISCRA-C Scientific Project *IMG-DEN*. We define a HPC implementation of the proposed deep learning framework, taking advantage of a large ultrasound data set with 10K ultrasound images, and of the CINECA-Marconi100 cluster, exploiting both CPUs (IBM POWER9 AC922) and GPUs (NVIDIA Volta V100). Given a training data set, composed of raw images and the corresponding denoised images, we implement a parallel and distributed deep learning framework in TensorFlow2. Then, we define a batch file for the execution of the deep learning framework on the cluster, which specifies the number of nodes, CPUs, GPUs, and memory of the cluster. Through the proposed HPC framework, we train multiple networks with large data sets in a reasonable time for the target medical application, thus increasing the specialisation to anatomical districts, and consequently the accuracy of the deep learning framework. The HPC framework generates a network model that is stored and used for predicting the output results. Furthermore, we can improve the offline training with new data, a-priori and/or additional information on the input data (e.g., input anatomical district, noisy type/intensity, image resolution, acquisition methodology/protocol). The update of the existing training data set is always performed offline, through the addition of new images after expert validation of the denoising results.

Training data sets & metrics We consider the *Esaote private data set*, which contains more than 10K ultrasound images at different resolutions, and is acquired from different (e.g., muscle-skeletal, obstetric, abdominal) anatomical districts. In this data set, we compare the performance of different denoising methods applied to ultrasound images, and analyse the performance of the proposed framework, through the training and the prediction of the learning-based network, with a specialisation to anatomic districts. We generate and test different data sets, by varying the number of images for the training phase, and the anatomical district for the prediction phase. The custom Pix2Pix network is trained on four data sets of obstetric images, respectively: (a) 500, (b) 1500, (c) 3500, and (d) 5000 images. Each data set is composed of the input images (i.e., the raw Esaote data set) and the target images (i.e., the corresponding images denoised with the tuned-WNNM). The validation data set is composed of an additional set of different images (i.e., about 10% of the training data set) of the same district. Then, we evaluate each of the four networks (i.e., the networks trained with a different number of images) with two different test data sets of 50 images each, respectively from the (i) obstetric and (ii) muscular anatomical districts.

As *quantitative metrics*, we consider the *peak-signal-to-noise ratio* (PSNR) and the *structural similarity index measure* (SSIM) for the comparison of the raw input with the predicted denoised image provided by the proposed framework. The PSNR measures the mean squared error (MSE) through a logarithmic scale; the SSIM measures the perceived similarity of two images, calculated on different windows. Given two identical images, the PSNR and SSIM values are equal to $+\infty$ and 1, respectively. Furthermore, we integrate the quantitative metrics with a qualitative discussion on the quality of the denoised images, in terms of blurring and edge preservation.

The qualitative validation is performed by Esaote experts with experience in ultrasound image processing. The main parameters for the validation are the preservation of edges and anatomical geometries, the non-alteration of grey levels and image properties (e.g., brightness), the generation of artefacts (e.g., blurring, dragging effects) or patterns.

3.2 Real-time US 2D image denoising: experimental results

We present the results of denoising methods with a specialisation to ultrasound images (Sect. 3.2.1) and the prediction results of the proposed framework on US 2D images (Sect. 3.2.2).

3.2.1 Comparison of denoising methods

We evaluate the denoising results on different anatomical districts of the Esaote data set (Figs., 3.3, 3.4, 3.5): WNNM, NCSR, and PCA-BM3D have been judged as the best methods

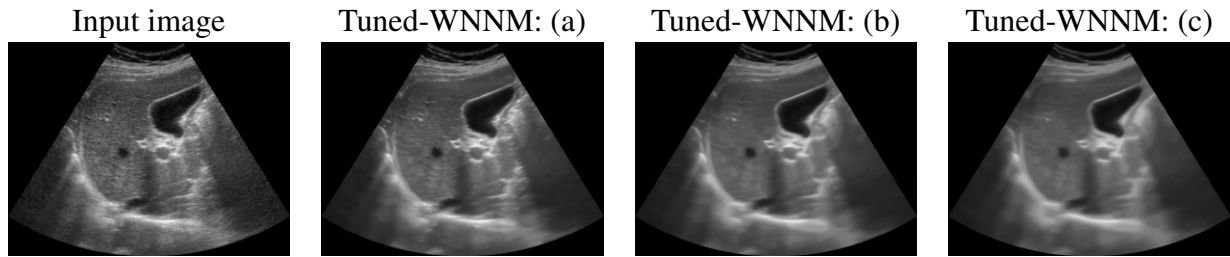


Figure 3.6: Ultrasound image of an abdominal district and denoised images achieved by applying the tuned-WNNM and varying the denoising intensity from low (a) to high (c).

in terms of denoising, and WNNM outperforms all the other methods in terms of edge preservation and enhancement. In particular, WNNM well preserves the edges of the muscular fibres (Fig. 3.3) and the internal organs (Figs. 3.4, 3.5). The output of SAR-BM3D shows a granular effect, which negatively affects the preservation of the anatomical features, and BM-CNN generates artefacts, which are typical of a deep learning approach. Finally, the discussed methods improve the results of the algorithm currently implemented at Esaote, which runs in real-time and moderately removes noise from the raw image. According to these results, we select WNNM as the best method for the denoising of ultrasound images. However, we underline that the other methods have their characteristics in terms of denoising and edge preservation, and they could be included in the framework as alternative denoising algorithms.

Then, we evaluate and compare the denoising results of *tuned-WNNM* (Figs., 3.3, 3.4, 3.5). Also, we analyse the role of the parameter that affects the threshold of the singular values of the SVD, and consequently the intensity of the smoothing. Increasing this parameter, the method improves in terms of removed noise, though introducing a low blurring effect. To select the best tuning for denoising intensity, we select the output image that best fits the medical requirements, among three different levels of denoising intensity (Fig. 3.6). In particular, Fig. 3.6(b) shows the best result as a compromise between noise removal, edges preservation, and blurring effect; in fact, it preserves the geometry of the internal tissues, while enhancing the edges of the anatomical structures. Additional analysis is presented in the experimental tests and validation with synthetic data (Sect. 5.1).

Execution time Our tests (Table 3.1) are executed with Matlab R2020a, on a workstation with 2 Intel i9-9900KF CPUs (3.60GHz), 32 GB RAM, and none of these methods achieves real-time computation. In particular, WNNM takes more than three minutes to process a 600×485 image, and the fastest method (i.e., SAR-BM3D) takes about one minute; however, real-time computation in an ultrasound environment requires a processing time in the order of a few milliseconds. This result motivates the proposed development of a deep learning framework for the real-time denoising of ultrasound images, further optimised with a HPC framework (Sect. 3.1).

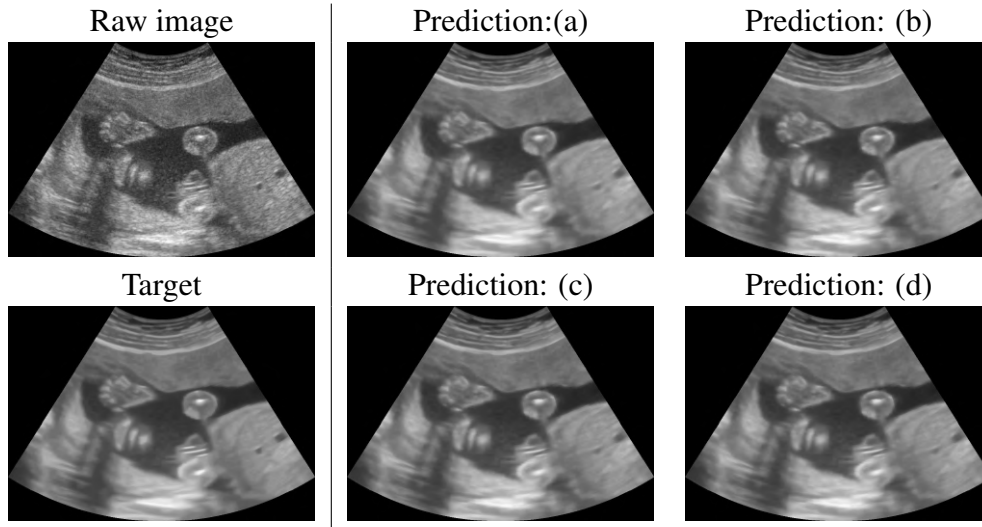


Figure 3.7: Raw, target, and prediction images, related to the obstetric data set (i). Training set: (a) 500 images, (b) 1500 images, (c) 3500 images, (d) 5000 images (Sect. 3.1).

Increasing this parameter, the method improves in terms of removed noise, though introducing a low blurring effect. To select the best tuning for denoising intensity, we select the output image that best fits the medical requirements, among three different levels of denoising intensity (Fig. 3.6). In particular, Fig. 3.6(b) shows the best result as a compromise between noise removal, edges preservation, and blurring effect; in fact, it preserves the geometry of the internal tissues, while enhancing the edges of the anatomical structures.

3.2.2 Framework prediction results

We present the prediction results of our framework through a quantitative and qualitative evaluation, analysing the properties of the training data set and the learning architecture, and discussing the execution time and computational cost.

Qualitative results Regarding the deep learning framework, and the large ultrasound data set (Sect. 3.1), Fig. 3.7 shows the prediction results of the four networks, when tested with obstetric images (i). The predicted images are very close to the target image in all four cases; the edges and the grey-scale values are well reproduced by the network. Furthermore, the predictions do not generate artefacts or spurious patterns. Varying the number of images of the training data set from 500 to 5K (Figs. 3.7(a-d)), the predicted images are slightly better than the target denoised images. Nevertheless, the results are good even with a small training data set of 500 images. Fig. 3.8 shows the prediction results of the four networks when tested with muscle-skeletal images (ii). Predicting the output images with the networks trained with obstetric images

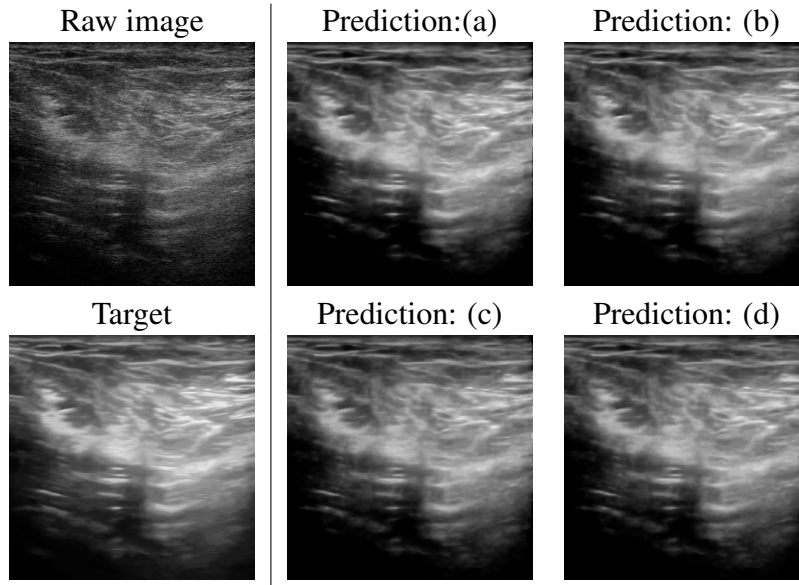


Figure 3.8: Raw, target, and prediction images, related to the muscle-skeletal data set (ii). Training set (Sect. 3.1) with (a) 500, (b) 1500, (c) 3500, and (d) 5000 images.

(Figs. 3.8(a-d)), the results are slightly worse than the corresponding case in Fig. 3.7, even if the predicted images do not show any artefact of pattern repetition. These networks are trained with images from a different (i.e., obstetric) district, with different anatomical features. This result confirms that each district requires a specific network and that a single network for all the districts gives lower quality results.

Quantitative results Table 3.2 reports the quantitative metrics (Sect. 3.1) computed between the target and the predicted images. The network trained with 5K images (d) tested with obstetric images (i) has a median PSNR and SSIM value of 36.13 and 0.964, respectively, while the same network tested with muscle-skeletal images (ii) has a median PSNR and SSIM value of 26.58 and 0.881. Both the metrics have a very slight improvement when passing from a training set of 500 to a training set of 3500 images, confirming the results of the qualitative analysis. An additional increase of the training data set size to 5K images further improves the quantitative results for both the test data sets. Fig. 3.9 shows the box plot of the PSNR and SSIM metrics for four training data sets and two test data sets. Increasing the number of images of the training data set, the range of the metrics tends to decrease; this behaviour has a lower variability on the prediction of the output image. These results confirm that a network specialised in a single anatomical district reaches the best denoising quality. The prediction of muscle-skeletal images from a network trained with obstetric images highly reduces the performance of our framework; in fact, the network learns that replicates not only the denoising algorithm itself but also its adaptation to the anatomic structures and features of each district.

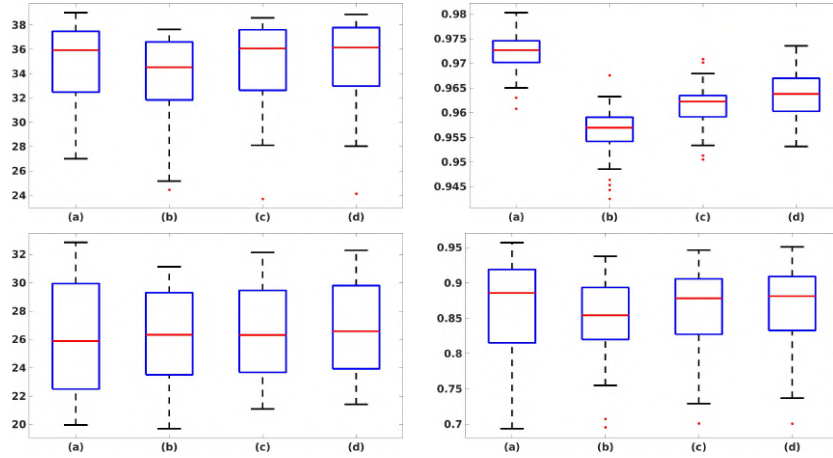


Figure 3.9: PSNR and SSIM boxplot for each of the four training data set (i.e., (a-d)) and the two test data sets (Sect. 3.1): (top-left) PSNR, obstetric test data set; (top-right) SSIM, obstetric test data set; (bottom-left) PSNR, muscle-skeletal test data set; (bottom-right) SSIM, muscle-skeletal test data set.

Table 3.2: Concerning the four training data sets and the two test data sets (i.e., obstetric: *Ob.*, and muscle-skeletal: *Msk.*) described in Sect. 3.1, we report the PSNR and SSIM metrics computed between the target and the prediction images, as median value among the 50 test images.

| Metrics | PSNR | | SSIM | |
|----------------------|-------|-------|-------|-------|
| | Ob. | Msk. | Ob. | Msk. |
| Test data set | | | | |
| Training with | | | | |
| (a) 500 images | 35.93 | 25.88 | 0.973 | 0.886 |
| (b) 1500 images | 34.52 | 26.33 | 0.957 | 0.854 |
| (c) 3500 images | 36.07 | 26.31 | 0.962 | 0.878 |
| (d) 5000 images | 36.13 | 26.58 | 0.964 | 0.881 |

Single versus multiple districts We compare the prediction results of our framework with three different training data sets. The first two data sets have 500 and 1500 ultrasound images of the same district (i.e., the obstetric one), respectively. The third one is composed of 1500 images of different districts; in particular, we select 500 images from the cardiac, obstetric, and muscle-skeletal districts. Due to the different resolutions of the images, padding has been applied to obtain the same input resolution for each network. We evaluate the prediction results on four test data sets: the first three are composed of 50 images from the cardiac, obstetric, and muscle-skeletal districts, respectively. The fourth is composed of 50 images randomly selected from the aforementioned three districts. The prediction results (Fig. 3.10 and Table 3.3) show that the networks trained with obstetric images (i.e., the single district networks) give the best results with the obstetric test data set: the predicted image of the single district network shows fewer scattering artefacts than the multiple districts network. Also, the single district networks have

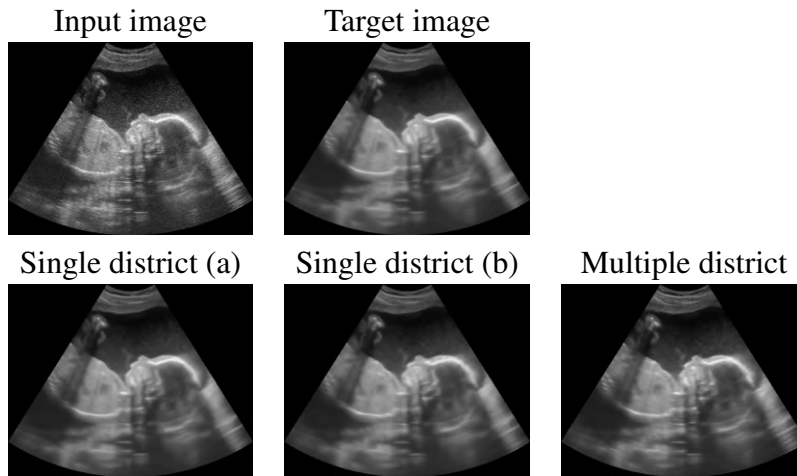


Figure 3.10: Prediction results of the obstetric district, with the networks trained with 500 (a) and 1500 (b) images from the obstetric district, and 1500 images from multiple districts (500 obstetric, 500 cardiac, 500 muscle-skeletal images).

better results in terms of quantitative metrics: adding further images from different districts to the training data set worsens the results; in fact, the single district network with 500 obstetric images has a PSNR value of 35.93, while the multiple districts network has a PSNR value of 33.70.

Comparing the results on the other test data sets (e.g., cardiac Fig. 3.11 and Table 3.3), the network trained with images of multiple districts has better results than the networks trained with obstetric images only. The multiple district network better generalises on the denoising algorithm, more than on the anatomic features, thus generating fewer artefacts on the prediction. Furthermore, the multiple district network includes 500 cardiac images in its training data set, thus improving the prediction results in this district. These features allow the multi-district network to reduce the artefacts on the edges of the anatomical features. A similar result is present in the muscle-skeletal test data set, where the results of the network trained with multidistrict images generalise better than the network trained with obstetric images; in fact, the single district network with 1500 obstetric images has a PSNR value of 26.33, while the multiple districts network has a PSNR value of 28.41. As the main conclusion, a dedicated network for each anatomic district is the best solution for the prediction of the denoised ultrasound images of each specific district, if a sufficiently large data set is available for the training.

Deep learning architectures We compare the prediction results of two different networks: Pix2Pix and the (*Matlab*) CNN, as part of our deep learning framework. Fig. 3.12 shows that Pix2Pix has better results than CNN, in terms of blurring reduction, noise removal, and edge preservation. We also compare the quantitative metrics between the target images and the pre-

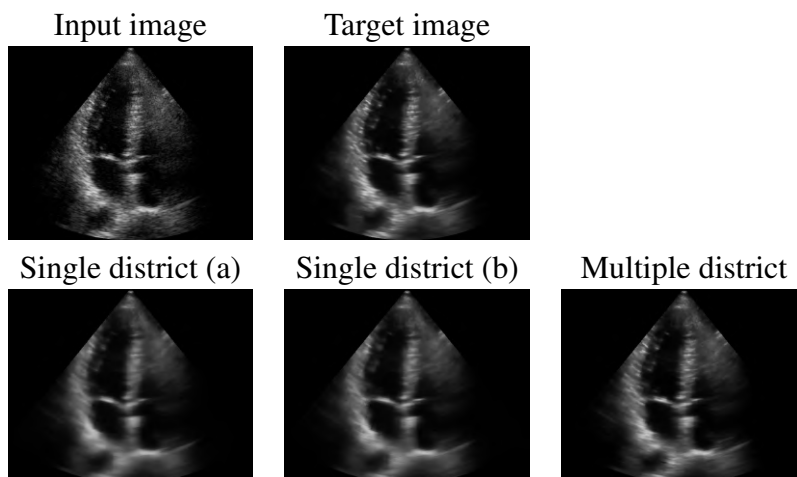


Figure 3.11: Prediction results of the cardiac district, with the networks trained with 500 (a) and 1500 (b) images from the obstetric district, and 1500 images from multiple districts.

Table 3.3: Concerning the results in Figs. 3.10, 3.11, we report the PSNR metric computed between the target and the prediction images, as the average value among the 50 images of each test data set: obstetric (Ob.), muscle-skeletal (Msk.), cardiac (Card.), and multiple districts (Multi.). The networks are trained with: single district (a, 500 obstetric images), single district (b, 1500 obstetric images) and multiple district images.

| Test data set | Ob. | Msk. | Card. | Multi. |
|----------------------|-------|-------|-------|--------|
| Training with | | | | |
| Single district (a) | 35.93 | 25.88 | 28.46 | 29.47 |
| Single district (b) | 34.52 | 26.33 | 28.63 | 29.51 |
| Multiple district | 33.70 | 28.41 | 33.82 | 30.33 |

dicted images, on the test data set of 50 ultrasound images of the obstetric anatomic district (Sect. 3.1). Pix2Pix has a PSNR average value of 36.07, and an SSIM average value of 0.878, while CNN has a PSNR average value of 25.69, and an SSIM average value of 0.651. This result underlines that Pix2Pix outperforms CNN as network architecture for our deep learning framework.

Execution time and computational cost To test the training phase of the deep learning framework on the HPC environment, we exploit 8 nodes, each one composed of 32 cores and 4 accelerators, for a theoretical computational performance of 260 TFLOPS, and 220 GB of memory per node. The parallel implementation of the deep learning framework and the high hardware performance reduce the computation time of the training phase by at least 100 orders less than a serial implementation on a standard workstation.

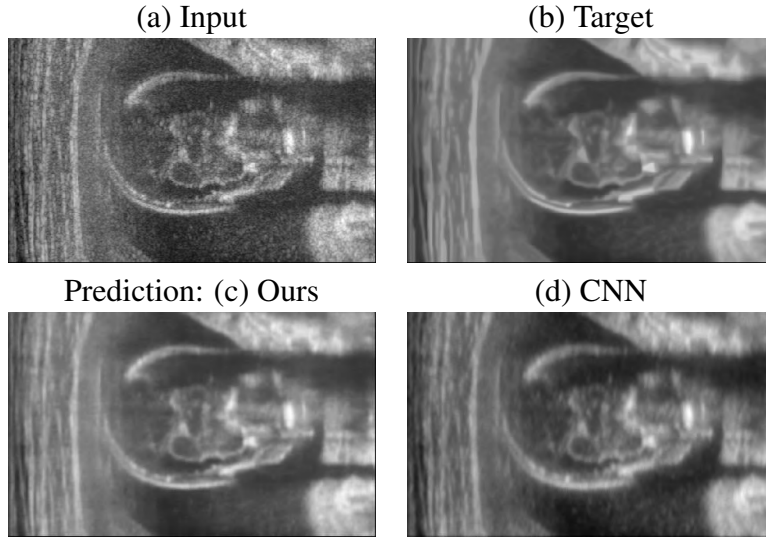


Figure 3.12: (a) Input, (b) target, (c) our prediction with Pix2Pix, and (d) prediction with CNN, for the obstetric district.

The prediction’s execution time is crucial for our framework’s real-time implementation. We test the denoising prediction on GPU-based hardware, which replicates the hardware of an ultrasound scanner currently in use. Given a set of ultrasound input images from different districts, the average execution time is 25 milliseconds; this result confirms that we achieve the real-time computation target, required by the industrial constraint. After the industrialisation of the proposed framework, the execution time can be further reduced with a hardware-oriented implementation that optimises the prediction operations of the neural network in terms of computation.

We underline that the input resolution of the network is 600×600 , which is reached through the zero-padding of each input image. The computational cost of the prediction depends on the resolution of the input image and on the architecture of the network: in particular, the computational cost of a convolution operation is $\mathcal{O}(r/s_r \cdot c/s_c) \cdot (f_r \cdot f_c) \cdot f$; in our application, the input image has a resolution of $r = c = 600$, the kernel-filter size on rows and columns is $f_r = f_c = 4$, the stride on rows and columns is $s_r = s_c = 2$, we use 10 convolution and 10 deconvolution operators, and a number of kernel-filters from 32 to 512.

3.3 Learning-based low-rank denoising

An input image $\mathbf{Y} = f(\mathbf{X}, \mathbf{N})$ is composed of the ground truth \mathbf{X} and the noisy component \mathbf{N} , where f defines the combination of one or more type of noise (e.g., additive, impulsive, multiplicative). Low-rank approximations recover the underlying low-rank matrix from its degraded observation. In this class, the *nuclear norm minimisation* recovers an estimation $\hat{\mathbf{X}}$ of the ground

truth signal through the minimisation of the energy functional $\|\mathbf{Y} - \mathbf{X}\|_F^2 + \lambda\|\mathbf{X}\|_*$, where the *nuclear norm* $\|\mathbf{X}\|_*$ is the sum of the singular values of \mathbf{X} and measures the compliance of the approximated image to the high-frequency components of the input image. This problem is equivalent to recover the approximated image $\hat{\mathbf{X}} = \mathbf{U}(\tau(\mathbf{S}))\mathbf{V}^\top$, where $\mathbf{Y} = \mathbf{U}\mathbf{S}\mathbf{V}^\top$, $\mathbf{S} := \text{diag}(\mathbf{s})$, is the SVD of the noisy signal, and $\tau(\mathbf{S}) = \text{diag}(\mathbf{s} - \lambda)$ with λ constant vector. The thresholding $\tau(\mathbf{S})$ improves the image reconstruction by removing the noise component while preserving image features. Weighted thresholds imply a different shrinkage to each singular value, as $\mathbf{S}_{hh} - \lambda_h$, where \mathbf{S}_{hh} is the (h, h) entry of \mathbf{S} . The proposed learning-based method trains a network to predict the optimal thresholds λ_h . We present the computation of the optimal thresholds and the related denoising accuracy (Sect. 3.3.1), the setting and optimisation (Sect. 3.3.2) of the learning model for the prediction of the thresholds, and the experimental results on US 2D images (Sect. 3.3.3).

3.3.1 Denoising with optimal threshold values

The computation of optimal thresholds through a minimisation problem allows us to apply the optimal weights to the weighted nuclear norm minimisation problem and set an upper limit to the accuracy of the denoising through the method introduced in Eq. (2.2).

Optimisation model for the computation of the optimal thresholds Given a $n \times n$ ground truth image \mathbf{X} and a noisy instance $\mathbf{Y} = f(\mathbf{X}, \mathbf{N})$, we compute the optimal thresholds λ_h through the minimisation of the distance between the ground truth and the reconstructed signal $\hat{\mathbf{X}}$, as

$$E(\lambda) = \sum_{i=1}^n \sum_{j=1}^n |\mathbf{X}_{ij} - \hat{\mathbf{X}}_{ij}|^2, \quad (3.1)$$

where the reconstructed signal is computed through the SVD of the \mathbf{Y} signal with weighted thresholds, i.e., $\hat{\mathbf{X}}_{ij} = \sum_{h=1}^n \mathbf{U}_{ih}(\mathbf{S}_{hh} - \lambda_h)\mathbf{V}_{hj}$. The first-order derivatives of the energy functional $E(\lambda)$ concerning the variables of the optimisation problem λ is

$$\frac{\partial E(\lambda)}{\partial \lambda_k} = 2 \sum_{i,j=1}^n \left[\mathbf{X}_{ij} - \sum_{h=1}^n \mathbf{U}_{ih}(\mathbf{S}_{hh} - \lambda_h)\mathbf{V}_{hj} \right] \mathbf{U}_{ik}\mathbf{V}_{kj}.$$

The minimum of Eq. (3.1) is computed through the iterative optimisation method L-BFGS (*Limited-memory Broyden, Fletcher, Goldfarb, Shanno*) [ZBLN97], which finds the roots of the derivative of the energy functional. L-BFGS is an optimisation algorithm in the family of quasi-Newton methods that approximates the Broyden-Fletcher-Goldfarb-Shanno algorithm (BFGS) using a limited amount of computer memory. Analogously to BFGS, the L-BFGS solver estimates the inverse Hessian matrix for the minimum search in the variable space; however, the

L-BFGS method represents the approximation implicitly through a few vectors, thus involving a limited memory requirement. At each iteration, a small history of the past updates of the variables (λ) and the gradient of the energy functional $E(\cdot)$ in Eq. (3.1) is used to identify the direction of steepest descent and to implicitly perform operations requiring vector products with the inverse Hessian matrix. For the L-BFGS method, the memory storage is $\mathcal{O}(u^2)$ and the computational cost is $\mathcal{O}(uv)$ at each iteration, where u is the total number of variables, and v is the number of steps stored in memory.

Application of optimal thresholds for denoising The direct application of the optimal thresholds to the input noisy image has reduced accuracy, due to the complexity of the geometries of the image and the difficulty in distinguishing the edges and structures from the noise components. Therefore, the introduction of a *block-matching* algorithm allows us to compute stacks of similar patches with redundancy properties and apply the SVD and the optimal thresholds to these stacks, where our method can identify and then remove the noisy component, maintaining the geometric and texture structures. Furthermore, the application of the optimal thresholds through an *iterative approach* allows for progressively improve the accuracy of the solution.

Block-matching Given a pixel y , the *patch* P_y is the set of pixels in the neighbourhood of y . The *search window* is the set of selected patches used to search the closest ones to a reference patch, according to certain metrics. The *stack*, or *3D block*, is the set of patches similar to a reference patch. The block-matching algorithm stacks similar patches into a 3D structure and uses the redundancy of the stack to remove the noise. Then, the patches are aggregated by a collaborative filter to compose the denoised image.

Iterative approach We compute the optimal denoising of three images with a different noise. The method is applied for 20 iterations: at each iteration, the denoised image of the previous iteration is used as a noisy image, while the ground truth is fixed. We compute the 3D blocks with the block-matching algorithm and extract their ground truth counterparts from the ground truth image. Then, we compute the optimal thresholds λ_h for the SVD through the minimisation of Eq. (3.1) for each 3D block, and we reconstruct the denoised image. The results are discussed in the validation on synthetic images (Sect. 5.3).

3.3.2 Learning model: setting and optimisation

We design the data set and the learning model for the computation of the optimal thresholds.

Training & clustering data set Applying an arbitrary noise to a set of input images, we build a data set of ground truth and noisy images to train a learning-based method to predict the optimal thresholds. In particular, the input of the network is the vector of the singular values of the noisy block and the target is the vector of the optimal thresholds (Eq. (3.1)).

Instead of defining a single data set for all the images, we organise the images according to four parameters: (i) *type of noise*, (ii) *iteration*, (iii) *noise intensity*, and (iv) *singular values clustering*. This classification improves the specificity of the network and the accuracy of the prediction of the thresholds. The four parameters are nested: for each image noise (i) and iteration step (ii), we compute the 3D block's noise intensity (iii) with the *nonlinear noise estimator* (NOLSE) [LTF13] metric and define c intervals so that the NOLSE values are uniformly distributed into c classes. We cluster the singular values (iv) into k groups with the *k-means* algorithm; then, we train a Support Vector Machine (SVM) [CST00] to classify the vectors of singular values according to the *k-means* labels. For each combination of the four parameters, we define the matrix $\mathbf{P} = \{\mathbf{P}_k\}_{k=1}^N$, with $\mathbf{P}_k = [p_1, \dots, p_n]^\top$, where N is the number of 3D blocks and n is the lower dimension of the 3D block; the p elements are the singular values computed through the SVD. These values are associated with the corresponding optimal thresholds $\mathbf{Q} = \{\mathbf{Q}_k\}_{k=1}^N$, computed as the minimisation of Eq. (3.1).

This classification is applied to the training of the learning-based networks to improve the specialisation of each prediction; the training is performed on synthetic images (Sect 5.3) where the ground truth is available. The trained network is applied to predict the optimal denoising for US images (Sect. 3.3.3); in this context, we apply our learning-models trained on images with speckle noise at different intensities; then, we apply five iterations with the respective trained network, computing the noise intensity and *k-means* prediction for each block.

Optimisation model For each combination of the four parameters (Sect. 3.3.2), we compute a dedicated learning-based network. Given the singular values of a 3D block, we train a network to predict the thresholds that reconstruct the best approximation of the ground truth block. The network weights are defined through the matrix \mathbf{W} of dimension $n \times n$, and are computed by minimising the distance between the target and the predicted thresholds, as

$$F(\mathbf{W}) = \sum_{k=1}^N \sum_{j=1}^n \left| \mathbf{Q}_{jk} - \hat{\mathbf{Q}}_{jk} \right|^2, \quad (3.2)$$

where $\hat{\mathbf{Q}}_{jk} = \sum_{i=1}^n \mathbf{W}_{ij} \mathbf{P}_{ik}$ are the predicted thresholds. The first-order derivatives of Eq. (3.2) are

$$\frac{\partial F(\mathbf{W})}{\partial \mathbf{W}_{lm}} = -2 \sum_{k=1}^N \left(\mathbf{Q}_{mk} - \sum_{i=1}^n \mathbf{P}_{ik} \mathbf{W}_{im} \right) \mathbf{P}_{lk}, \quad (3.3)$$

and the second-order derivatives are

$$\frac{\partial^2 F(\mathbf{W})}{\partial^2 \mathbf{W}_{lm}} = 2 \sum_{k=1}^N |\mathbf{P}_{lk}|^2.$$

Since the functional in Eq. (3.2) is quadratic and its second-order derivatives are always strictly positive, the functional $F(\cdot)$ is strictly convex, and it has a unique minimum.

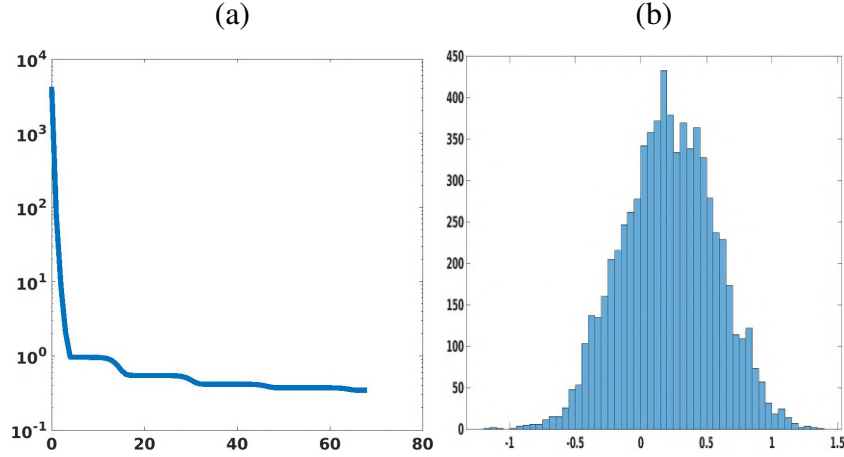


Figure 3.13: (a) Minimisation of the functional (Eq. 3.2) value (y -axis) normalised on the number of vectors $N = 200$, with respect to the number of iterations (x -axis); (b) frequency (y -axis) of the difference between the optimal and the predicted thresholds (x -axis) of the first singular value of the SVD, for all the blocks of the cameraman image.

Fig. 3.13(a) shows the convergence of the minimisation to the optimal solution computed with the L-BFGS algorithm (Sect. 3.3.1). The solver takes 67 iterations to converge to the minimum of the functional. Fig. 3.13(b) shows the frequency of the error between the optimal and the predicted thresholds for the first singular value $\#_i\{x \in (\mathbf{Q}_{1k} - \hat{\mathbf{Q}}_{1k}) : h_i \leq x \leq h_{i+1}\}$, with h bins of the histogram. The average error is 0.3 and the upper and lower bounds of the error are ± 1.5 .

The minimisation of the functional in Eq. (3.2) is equivalent to the solution of the linear system $\nabla F(\mathbf{W}) = (\mathbf{P}\mathbf{P}^\top)\mathbf{W} - \mathbf{P}\mathbf{Q}^\top = \mathbf{0}$ (c.f., Eq. (3.3)), with $n \times n$ coefficient matrix $\mathbf{P}\mathbf{P}^\top$ and $\text{rank}(\mathbf{P}\mathbf{P}^\top) = n$ when N is sufficiently greater than n . Increasing the number N of vectors of singular values (i.e., the columns of \mathbf{P} and \mathbf{Q}) from 250 to 2500 (Fig. 3.14a), the condition number of the coefficient matrix passes from 2.3×10^{10} to 1.5×10^{10} ; given the solution $\widetilde{\mathbf{W}}$, the approximation error of the linear system $\|\nabla F(\widetilde{\mathbf{W}})\|_F/N$ passes from 8.3×10^{-12} to 8.4×10^{-11} (Fig. 3.14b).

3.3.3 Experimental results on US 2D images

The main results of the training and prediction of the learning model are presented in the synthetic validation results (Sect. 5.3). We present the results of the application of the denoising method to ultrasound images.

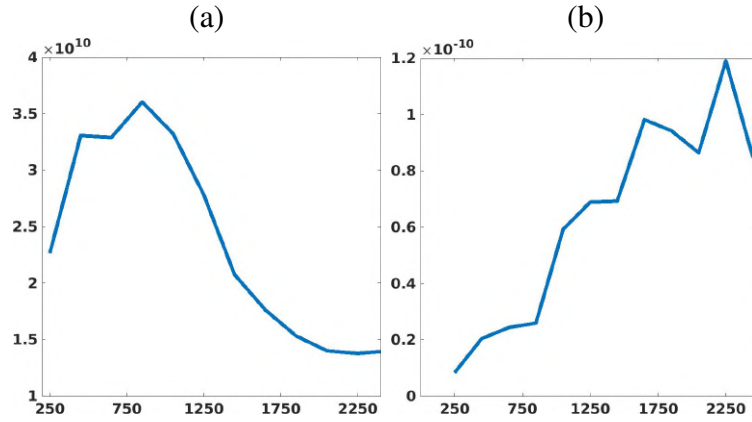


Figure 3.14: (a) Conditioning of the coefficient matrix (y -axis) with respect to N (x -axis). (b) Approximation error of the linear system (y -axis) with respect to N (x -axis).

Experimental results We compare our denoising method with previous work, on ultrasound images from the different anatomical districts (Fig. 3.15). In particular, we apply our learning-models trained on images with speckle noise at different intensities: $\sigma = 0.05$ (Fig. 3.15, ours(i)), and $\sigma = 0.10$ (Fig. 3.15, ours(ii)). The learning model trained with images with higher noise applies larger thresholds to the singular values of the SVD, resulting in more intense smoothing.

Our method correctly preserves the edges of the obstetric district (Fig. 3.15a) and the mass of the abdominal districts (Fig. 3.15b) while removing the noise component; this result is perceptible in the magnification, comparing our method with previous work.

Computation time The execution time for denoising each 3D block is 0.015 seconds; this time is composed of the noise intensity computation through the NOLSE metric (66%), the SVD (5%), the SVM prediction of the label of the singular values (14%), and the network prediction of the optimal thresholds (15%). Due to the operations performed for each 3D block, our method (330 secs.) increases the execution time for the denoising of a 256×256 image, compared to previous work: BM3D (1 sec.), WNNM (100 secs.), and NCSR (180 secs.). This execution time does not satisfy the real-time constraint for the Ultrasound application. However, our framework (Sect. 3.1) can learn the method and predict the output for real-time denoising of the US image.

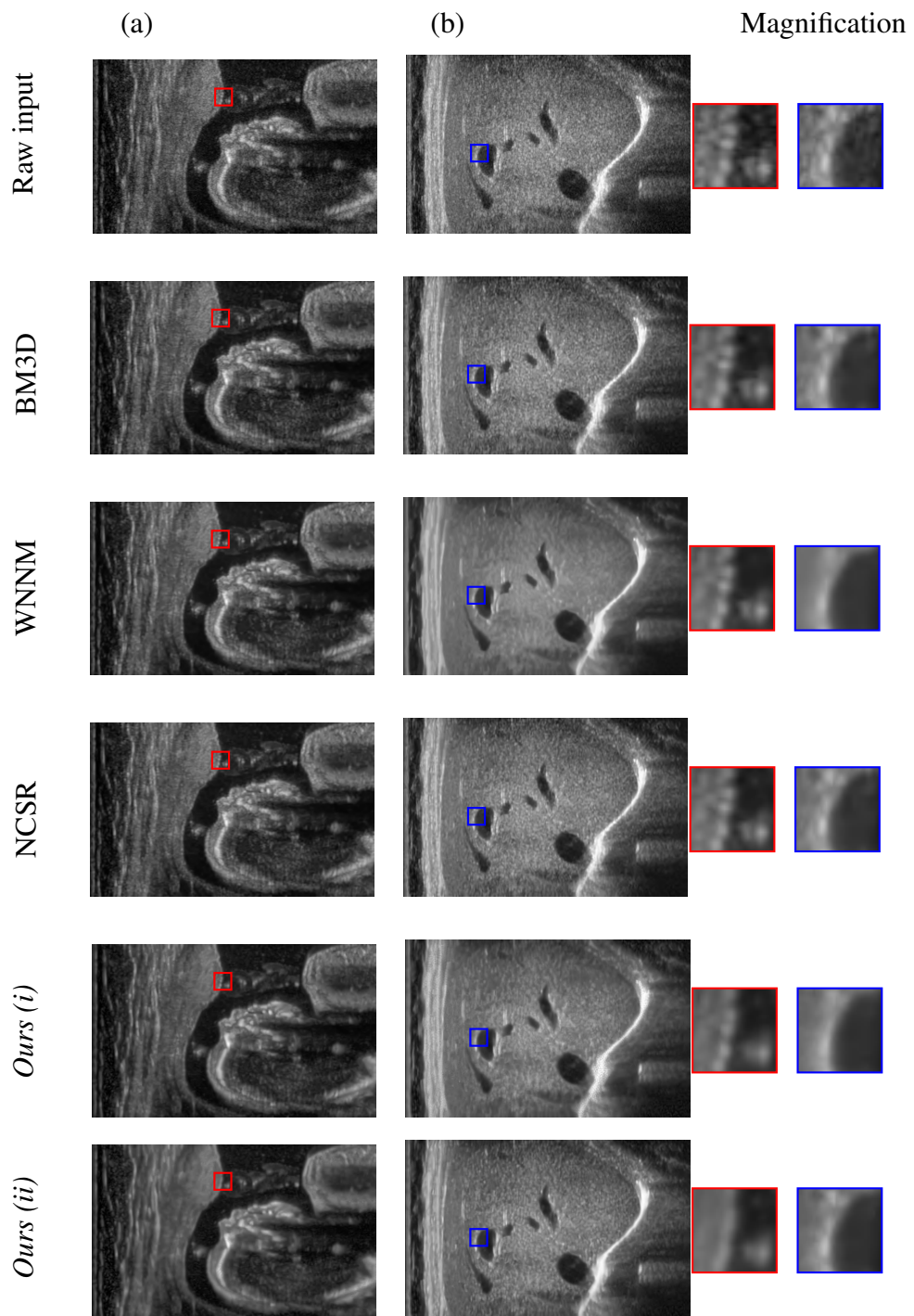


Figure 3.15: Comparison between previous work and our denoising with low (i) and medium (ii) smoothing intensity.

Chapter 4

Real-time denoising of US 2D videos and 3D images

Ultrasound images can be acquired at different planes, through the manual/automatic movement of the probe; furthermore, the physician can acquire both static and dynamic images. These acquisitions are associated with different input signals: 2D images, 2D videos, and 3D volumetric images. Each of these types of signals has its properties and advantages: dynamic 2D US images (i.e., videos) allow the physician to analyse the temporal variation of an anatomical feature (e.g., the movement of a muscle, the volume of the ventricle). This variation can be generated either by the shift of the probe or by the movement of the anatomical part. 3D volumetric images offer a full understanding of the spatial anatomy of the acquired region, against a more expensive tool for the acquisition (e.g., automatic 3D probes). US 2D images are simple to acquire and process, but they offer a limited view of the anatomical part of interest.

In this context, our goal is the development of a real-time method for denoising of ultrasound 2D videos and 3D images. We extend the framework proposed for US 2D images (Chapter 3), training an image-to-image learning network to replicate the behaviour of the denoising algorithms (e.g., BM4D, Vidosat). The network learns to predict images, which have been denoised through an algorithm that exploits the properties of the signal (e.g., temporal/spatial redundancy); then, the 2D videos / 3D images are reconstructed as ordered sequences of frames/slices. With this approach, we use a standard learning architecture, still exploiting the characteristics of the denoising algorithm. We also avoid the use of more complex architectures (e.g., spatio-temporal networks), and we avoid managing videos with different lengths for the training of the network. We specialise the training phase to specific anatomical districts and input signals. The real-time computation depends only on the execution time of the network prediction. As the main contribution, the proposed method is general in terms of the input data (e.g., 3D images, time-dependent US videos), type of noise (e.g., speckle, Gaussian noise), resolution (e.g., isotropic, anisotropic) of the input signals, acquisition methodology, and anatomical district.

For further details on the tests, we refer the reader to the video available at the URL: <https://www.dropbox.com/s/27rdwez8dibwdx/ICIAP21-HPC-3D-2DT.mp4?dl=0>

We review previous work on signals denoising and introduce the real-time denoising method for US 2D videos and volumetric images (Sect. 4.1) as an extension of the learning and HPC framework for 2D images (Chapter 3), and discuss the experimental results (Sect. 4.2).

Related publications

Cammarasana, Simone, Paolo Nicolardi, and Giuseppe Patanè. “*Fast Learning Framework for Denoising of Ultrasound 2D Videos and 3D Images*.”, Lecture Notes in Computer Science, Volume 13373 (2022). https://doi.org/10.1007/978-3-031-13321-3_42 [Conference Proceedings]

Cammarasana, Simone, Paolo Nicolardi, and Giuseppe Patanè, “*A General Framework for Smoothing Arbitrary Signals in Computer Graphics and Biomedicine*”, Arcachon, July 2022. https://cs2022.sciencesconf.org/data/pages/cs2022_abstracts.pdf [Extended Abstract]

4.1 Real-time denoising of US 2D videos and 3D images: proposed framework

The main requirements of a denoising algorithm for US signals are the magnitude of the removed noise, edge preservation, and real-time computation. The best results are achieved when a specific method is applied for the type of signal, a method that exploits the characteristics of the data (e.g., spatial, temporal redundancy, anisotropy). Video-based and volumetric-based methods well suit these requirements; however, the application of these methods requires that the signal be fully acquired before it can be processed, making the denoising through these methods intrinsically non-real-time. To achieve real-time computation and maintain good results in terms of denoising and edge preservation, we extend the deep learning and HPC framework proposed for 2D images to 3D images and 2D videos. In this application, the framework learns and replicates the dedicated denoising algorithms with an image-to-image prediction; this allows us to apply a denoising method which accounts for the properties of the specific signal, with real-time execution. The experimental tests on the CINECA Cluster are supported by the ISCRA-C Scientific Project *IMG-DEN*.

US Signals US signals include 2D videos of different anatomical districts (e.g., abdominal, cardiac, and obstetrical) and volumetric 3D images. 2D videos are acquired through a 2D probe,

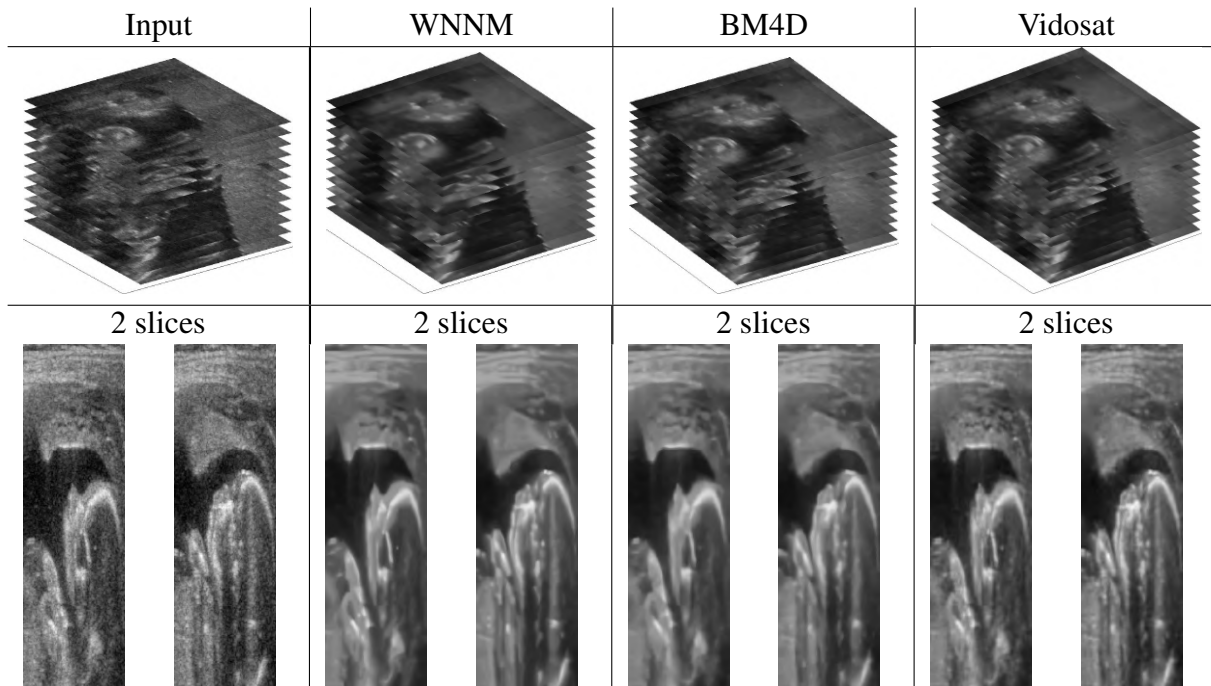


Figure 4.1: Denoising of US volumetric image $312 \times 146 \times 36$ of the obstetric district.

which captures sequences of images at a given frequency. The resolution of the images is affected by the acquisition frequency of the video since some anatomical districts (e.g., cardiac) require a high frequency to better analyse the behaviour of anatomical features that quickly change over time. Volumetric 3D images are acquired through probe oscillation within a given range; equally spaced slices are acquired and stacked into the 3D volume. Volumetric images are intrinsically anisotropic, due to the different resolution of each slice with respect to the resolution of the probe in its movement direction. We encode *2D images* as $m \times n$ matrices, *2D videos* as $m \times n \times t$ tensors, where each frame has a $m \times n$ resolution, and t frames are acquired. We represent *3D (volumetric) images* as $m \times n \times p$ tensors, where p planes are acquired, and each plane has a $m \times n$ resolution.

Proposed framework We extend the framework presented in Chapter 3 to the training of a neural network to learn and replicate the behaviour of the denoising algorithms applied to arbitrary signals. In the first phase, the denoising of different input signals is performed through dedicated algorithms (e.g., BM4D, Vidosat) which exploits the spatio-temporal redundancy. Then, the network is trained on a collection of pairs of US images: the input (i.e., the raw image) and the target (i.e., the image denoised with the proper filter). We train a specific network for each district, thus obtaining a more precise result when predicting the denoised image, as each network is specialised to the input anatomical features. Furthermore, images from different input signals (e.g., 2D videos, 3D images) are used to train specific networks. The real-time denoising

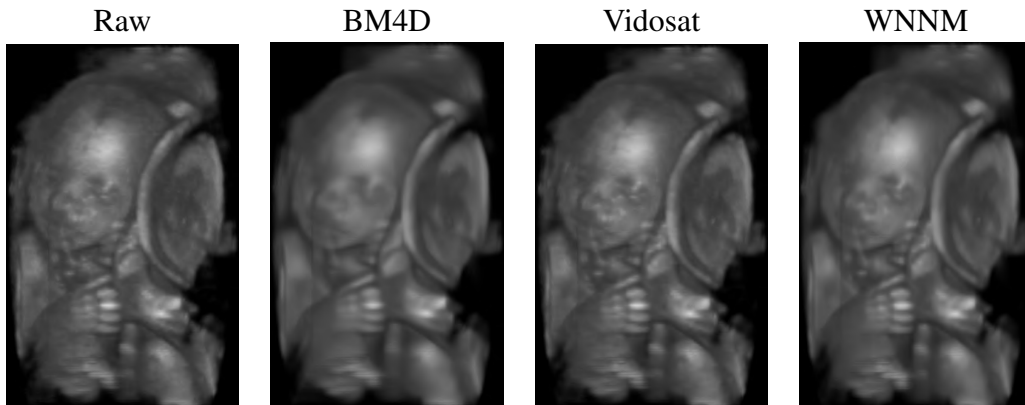


Figure 4.2: Ray-casting [Rot82] rendering of Fig. 4.1.

depends only on the execution time of the network prediction.

2D denoising methods WNNM is natively applicable to 2D images, and it is applied to 2D videos, through frame-by-frame denoising, and to 3D volumetric images, through slice-by-slice denoising. The application of WNNM to 2D videos/3D images does not exploit any knowledge of the temporal/spatial redundancy of the signal.

3D volumetric denoising methods BM4D is natively applicable to 3D volumetric images, which will be applied to 2D videos; in this case, the temporal dimension is assumed as the third dimension of the volumetric data. We follow two main approaches; given a 2D video $m \times n \times t$, where $m \times n$ is the resolution of a single frame, and t is the number of frames, BM4D can be applied to

- the full video: in this case, the $m \times n \times t$ tensor is the input video and the output is a tensor representing the smoothed video;
- consecutive portions of the video: in this case, the smoothing of each frame t_i is performed through the denoising of the tensor $m \times n \times 2k$, where $2k$ is the interval of frames centred in i , from $i - k$ to $i + k$. This procedure is repeated starting from the frame k , for each frame of the video, i.e., $t - k$ times.

2D video denoising methods *Vidosat* is natively applicable to 2D videos but can be applied to volumetric images, by considering the third spatial dimension as the temporal dimension of dynamic data. Both full video and consecutive portions (i.e., blocks of frames) can be considered as input of the denoising algorithm.

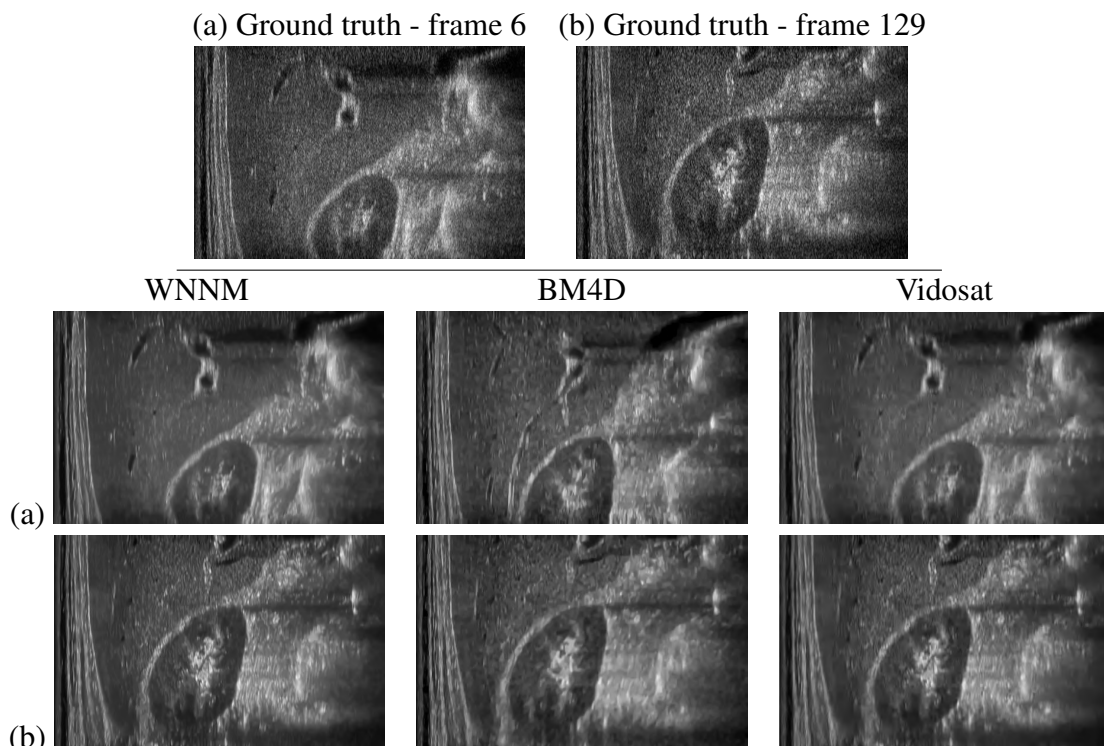


Figure 4.3: Denoising of 2 frames of a 2D US video ($560 \times 359 \times 256$) of the abdomen.

4.2 Real-time denoising of US 2D videos and 3D images: experimental results

We present the results of US 2D videos and 3D images denoising through dedicated algorithms (Sect. 4.2.1) and the prediction results of the proposed framework (Sect. 4.2.2).

4.2.1 Comparison of denoising methods on US signals

Fig. 4.1 shows the denoising results of WNNM, BM4D, and Vidosat, with *US volumetric images*. Analysing the images sectioned in the slice direction, WNNM and BM4D have better results in terms of noise removal and anatomical feature preservation, while Vidosat has a slight scattering effect. To integrate this evaluation, we apply a ray-casting algorithm [Rot82] to visualise a 2D rendering of the volumetric data (Fig. 4.2). This visualisation allows us to better analyse the consistency among slices; in fact, BM4D has a smoother rendering than WNNM, which is affected by a scattering effect, due to the phase shift between adjacent slices. Finally, Vidosat rendering is affected by the lower noise removal and BM4D has the best results in denoising 3D volumet-

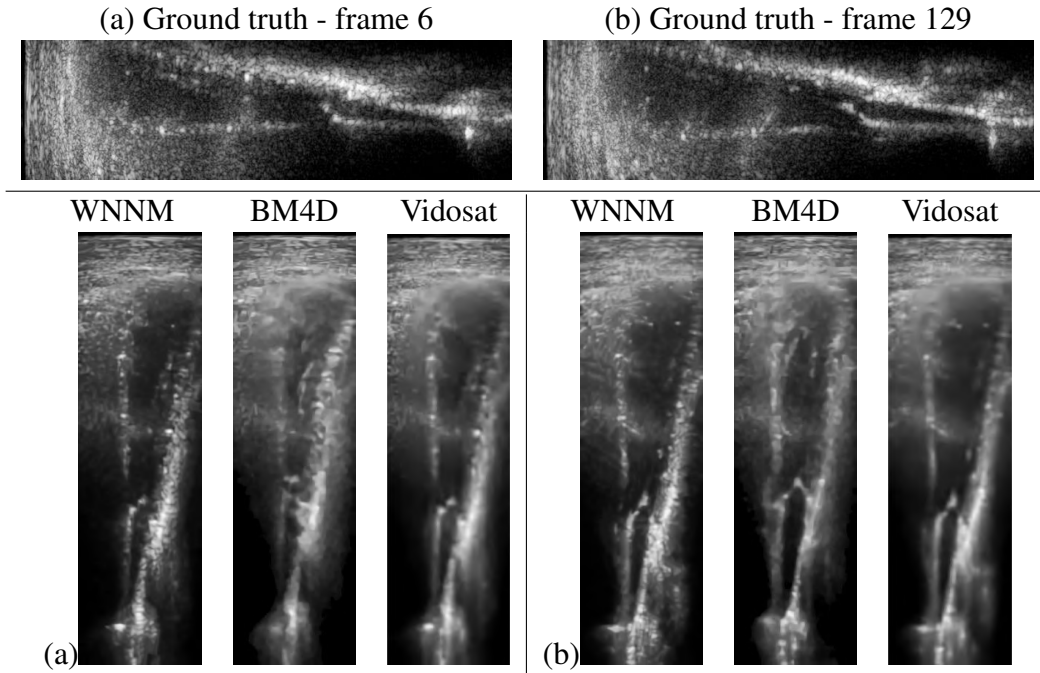


Figure 4.4: Denoising of 2 frames of a 2D US video ($592 \times 168 \times 233$) of the cardiac district.

ric images, in terms of noise removal, anatomical features preservation, and slice consistency. Figs. 4.3, 4.4, and 4.5 show the denoising results on *2D videos*, on different anatomical districts: abdominal, cardiac, muscle-skeletal. We select the window-moving version of BM4D, WNNM is applied *frame by frame*, and Vidosat is applied on the whole 2D video. Vidosat shows the best results in terms of noise removal and anatomical feature preservation; also, the blurring effect is barely present. WNNM is not affected by blurring effects, since it is applied frame by frame. However, the noise reduction of WNNM is lower than BM4D and Vidosat, since WNNM does not exploit any knowledge among consecutive frames. BM4D has a higher blurring effect and a lower noise reduction than Vidosat. Indeed, we select BM4D and Vidosat for generating the target images from 3D images and 2D videos, respectively.

4.2.2 Framework prediction results

We compare the qualitative, quantitative and execution time results of the proposed framework applied to US 2D videos and 3D images.

Qualitative results Our learning-based framework allows us to predict the denoised output of a 2D video (Fig. 4.6), and a 3D image (Fig. 4.7). In both cases, the predicted image is very similar to the target image; the output appears smooth, still preserving anatomic features and edges.

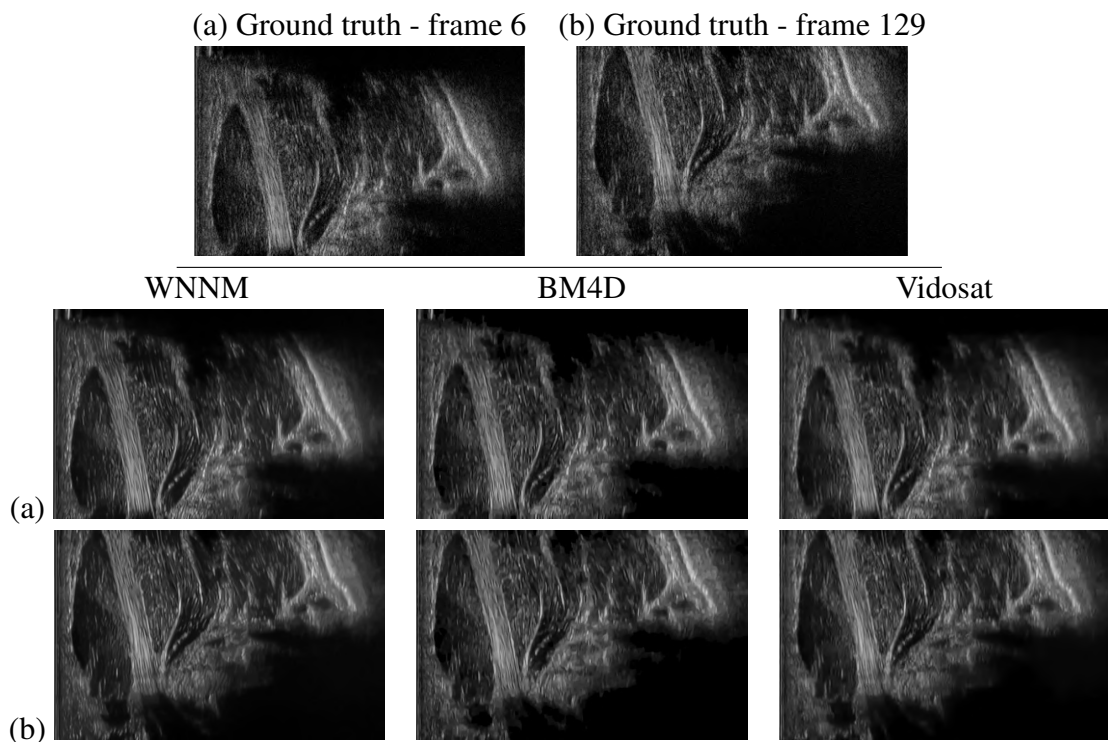


Figure 4.5: Denoising of 2 frames of a US video ($600 \times 380 \times 155$) of the muscle-skeletal district.

The predicted images do not present any artefacts or blurring effects. The smoothing intensity depends on the target images in the training data set; images with higher smoothing generate a network that learns higher denoising of the input image. Furthermore, our framework allows the user to train several networks with different levels of smoothing and to select the preferred output among different choices.

Quantitative error Table 4.1 summarises the quantitative error computed between the target and the predicted images, for different signals (e.g., 2D videos, 3D images) and anatomical districts (e.g., obstetric, cardiac), and measured through the PSNR and SSIM metrics. Images extracted from volumetric data and 2D videos are used to train separated networks. Each network is specific for an anatomical district (e.g., obstetric, cardiac). The selection of the anatomical regions depends on different parameters: the type of probe, the acquisition modality, the quantitative/qualitative analysis (e.g., segmentation, morphological evaluation), and the post-processing algorithms. Furthermore, different anatomical districts lead to different types of signals: for example, in the cardiac district, the movement of the cardiac valve is acquired by the probe, while in the abdominal district, the movement of the probe is performed by the physician. The experimental results show that our framework can correctly denoise the signal through an image-to-image prediction, for both 3D images and 2D videos. For example, the prediction of obstetric

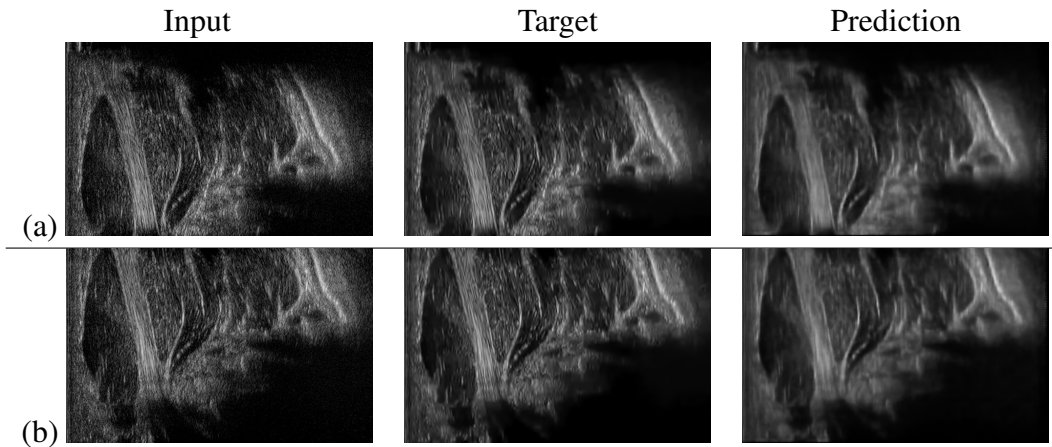


Figure 4.6: Framework prediction for 2 muscle-skeletal frames (a,b) of a 2D video.

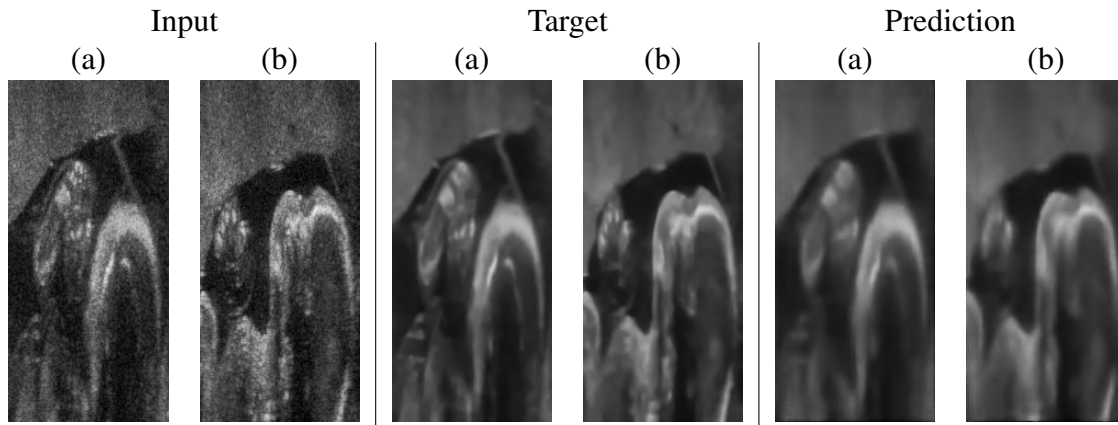


Figure 4.7: Framework prediction for two slices (a,b) of a 3D image of the obstetric district.

3D images shows an average PSNR value of 33.15, while the prediction of cardiac videos shows an average PSNR value of 34.62.

Execution time and computational cost Table 4.2 shows the *execution time* of the denoising methods, with different data set (i.e., 2D video, volumetric image). BM4D is the fastest method, taking 75 seconds for processing a $408 \times 120 \times 36$ volumetric image, while Vidosat and WNNM take 161 and 1260 seconds, respectively. None of these methods reaches the real-time requirement for denoising.

The analysis of the execution time of the framework is divided into an *offline training* and an *online prediction*. To test the training phase of the deep learning framework on the HPC implementation (Sect. 4.1), we exploit 8 nodes, each one composed of 32 cores and 4 accelerators,

Table 4.1: PSNR and SSIM metrics computed between the target and the prediction image, as average value among the 20 signals of each test data set: muscle-skeletal videos (Msk.), cardiac videos (Card.), obstetric 3D images (Ob.).

| Signal | Msk. | Card. | Ob. |
|---------------|-------|-------|-------|
| PSNR | 28.12 | 34.62 | 33.15 |
| SSIM | 0.963 | 0.978 | 0.970 |

Table 4.2: Concerning the tests in Figs. 4.1 and 4.5, we report the execution time of the denoising methods tested on 2D video and volumetric data.

| Method | WNNM | BM4D | Vidosat |
|---------------|-------|------|---------|
| 2D video [s] | 26820 | 1506 | 3338 |
| 3D image [s] | 1260 | 75 | 161 |

for a theoretical computational performance of 260 TFLOPS, and 220 GB of memory per node. The parallel implementation of the deep learning framework reduces the computation time of the training phase by at least 100 orders of magnitude compared to a serial implementation on a standard workstation, with an average execution time of fewer than 0.1 seconds per epoch on a single image, i.e., about 2 hours on a data set of 500 images, with 150 epochs. For the volumetric data set, we use 15 3D images composed of 36 stacks, for a total amount of 540 2D images. For the video data set, we use several videos of different lengths for each anatomical district; we train each network with a minimum amount of 500 2D images. All the images/videos are part of the private Esaote S.p.A. data set.

The execution time of the network prediction of our framework for the 2D image denoising is about 25 milliseconds, as already discussed in Sect. 3.2 and can be further optimised, thus confirming the real-time computation. The proposed framework allows the user to denoise each image during the acquisition of the video/volumetric signal since the spatio-temporal information is not required.

Chapter 5

Denoising: further experimental tests

The lack of ground truth signals is one of the main limits of the analysis of processing methods applied to biomedical data. The validation of the processing methods on synthetic signals allows us to have a quantitative measure of the accuracy of the method, comparing the available ground truth with the output of the processing method. In particular, synthetic signals allow us to compare the results of different methods, have an objective and reproducible comparison value, numerically compare parameter tuning (e.g., more or less intense smoothing), and present results and statistics aggregated over a large test data set. However, the use of synthetic data may not be fully representative of the biomedical image and the related anatomical structures, due to the different geometries and textures. Furthermore, the generation of synthetic noise, which is random, may differ from the noise generated by medical acquisition, which has physical significance. For these reasons, the clinical validation and the visual validation performed by experts in the medical sector remain the main methods of assessing the quality of a processing method on medical data.

5.1 Comparison among denoising methods on 2D images

We extend the comparison of the performance of the denoising methods (Sect. 3.2.1) on synthetic images (Sect. 5.1), and we compare WNNM with our tuned-WNNM method (Sect. 5.1).

Quantitative comparison We compare the five selected denoising methods on synthetic images, by adding speckle noise with different levels of noise intensity: given a noisy image $Y = X + NX$, where X is the normalised ground truth image, we define the artificial multiplicative noise $N(x) = \sqrt{12}\sigma u$, where $u \sim \mathcal{U}(-0.5, 0.5)$, \mathcal{U} is a uniform distribution, σ is the noise intensity, and x is a pixel of the image.

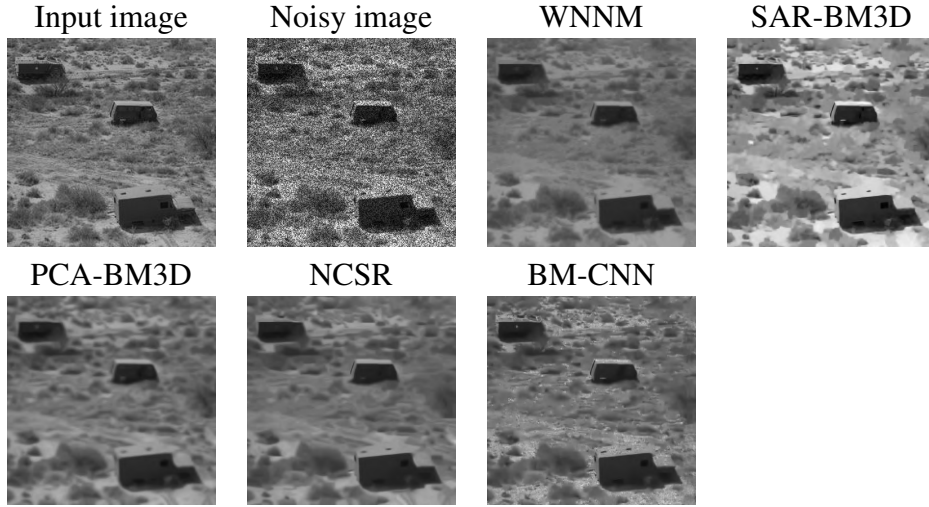


Figure 5.1: Input (SIPI data set, van image), noisy (speckle noise intensity $\sigma = 0.10$), and denoised images. For error metrics, we refer the reader to Table 5.1.

Table 5.1: PSNR and SSIM metrics of the denoising methods tested on the SIPI data set. For each σ value (i.e., the intensity of the speckle noise), we report the average metric computed on the 44 images of the data set.

| Metric | | PSNR | | | | SSIM | | | |
|---------------|----------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Method | σ | 0.05 | 0.1 | 0.2 | 0.3 | 0.05 | 0.1 | 0.2 | 0.3 |
| WNNM | | 25.57 | 24.68 | 23.35 | 22.32 | 0.681 | 0.659 | 0.630 | 0.602 |
| SAR-BM3D | | 27.36 | 26.01 | 24.71 | 23.65 | 0.730 | 0.699 | 0.673 | 0.651 |
| PCA-BM3D | | 25.09 | 24.36 | 23.05 | 22.10 | 0.652 | 0.640 | 0.614 | 0.585 |
| NCSR | | 26.60 | 25.36 | 23.61 | 22.36 | 0.665 | 0.669 | 0.619 | 0.588 |
| BM3D-CNN | | 26.85 | 24.20 | 20.21 | 17.26 | 0.733 | 0.569 | 0.31 | 0.215 |

The *SIPI data set* [Web97] is composed of 44 ground truth images of different sizes, organised in different classes (e.g., humans, landscapes). We evaluate the efficiency of the denoising methods: WNNM has very good results in terms of noise removal, edge preservation (e.g., vehicles shape (Fig. 5.1), and hat feathers (Fig. 5.2)). SAR-BM3D has the best results in terms of noise removal; however, it does not correctly preserve the grey-scale values (e.g., boy’s sleeve in Fig. 5.2) and it generates a blurred effect (e.g., grass and bushes in Fig. 5.1). PCA-BM3D and NCSR show minor preservation of edges and details than WNNM (e.g., boy’s face in Fig. 5.2). Finally, BM-CNN is not able to correctly remove the noise; this result underlines the importance of the training data set (e.g., the type and the intensity of the applied noise) when using a deep learning approach, and the necessity of using data-specific networks, instead of a generic-purpose one.

Table 5.1 summarises the results of the five denoising methods on the SIPI data set; we compute

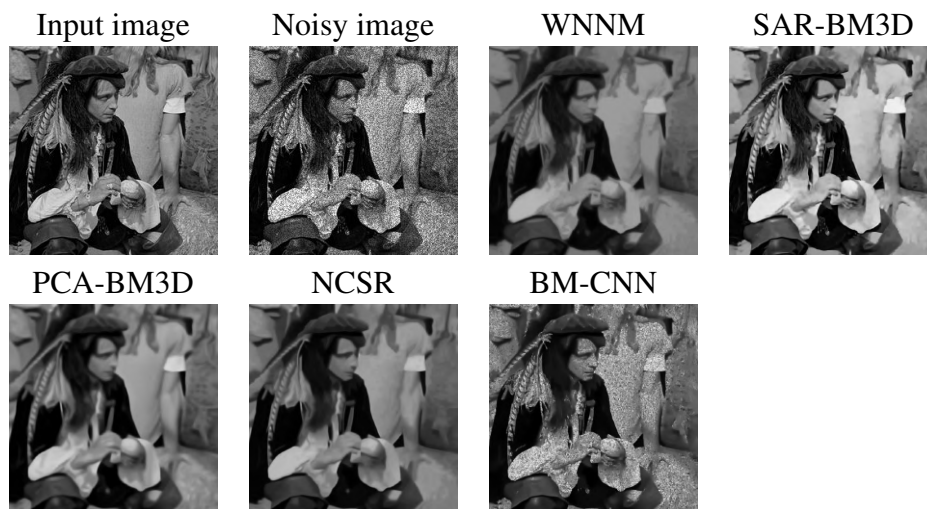


Figure 5.2: Input (SIPI data set, man image), noisy (speckle noise intensity $\sigma = 0.20$), and denoised images. For error metrics, we refer the reader to Table 5.1.

the average value of each metric (i.e., PSNR and SSIM) among 44 images of the data set, and report the average values when varying the intensity of the speckle noise. SAR-BM3D has the best results under these metrics, outperforming all the other methods. The NCSR, WNNM, and PCA-BM3D methods have good and similar results in terms of PSNR and SSIM indices. These four methods show a small degradation of the metrics values when increasing the noise intensity; this result is significant for ultrasound images, which generally have a different noise intensity, according to the anatomical district, the type of probe, and the data acquisition modality. Finally, BM-CNN shows a higher degradation of the PSNR and SSIM values, when increasing the noise intensity. Quantitative analysis is useful to compare methods with numerical measures, instead of performing only a visual evaluation. However, the main comparison among methods is the qualitative evaluation performed by the medical experts on ultrasound images, through the evaluation of the speckle noise removal and the preservation of anatomical features. We underline that, even if SAR-BM3D has better results than WNNM on synthetic images, WNNM has better performance on ultrasound images. Furthermore, our framework is general enough to use different denoising methods; two different learning networks can be trained, with WNNM and SAR-BM3D, to offer the physician the comparison between two denoising results.

WNNM vs. tuned-WNNM Comparing the baseline WNNM with the tuned-WNNM on synthetic images, we improve the denoising quality (Fig. 5.3) in terms of quantitative metrics; in fact, the output of WNNM has a PSNR value of 26.67, while the output of tuned-WNNM has a PSNR value of 26.74. Nevertheless, the execution time of WNNM is 94 seconds, while the execution time of tuned-WNNM is 260 seconds. We also compare tuned-WNNM and WNNM on the SIPI data set (Table 5.2). The aggregated results show that tuned-WNNM has slightly

Table 5.2: PSNR and SSIM metrics of WNNM and tuned-WNNM, tested on the SIPI data set. For each σ value (i.e., the intensity of the speckle noise), we report the average metric computed on the 44 images of the data set.

| Metric | | PSNR | | | | SSIM | | | |
|------------|----------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Method | σ | 0.05 | 0.1 | 0.2 | 0.3 | 0.05 | 0.1 | 0.2 | 0.3 |
| WNNM | | 25.57 | 24.68 | 23.35 | 22.32 | 0.681 | 0.659 | 0.630 | 0.602 |
| tuned-WNNM | | 25.60 | 24.76 | 23.49 | 22.61 | 0.685 | 0.663 | 0.642 | 0.627 |

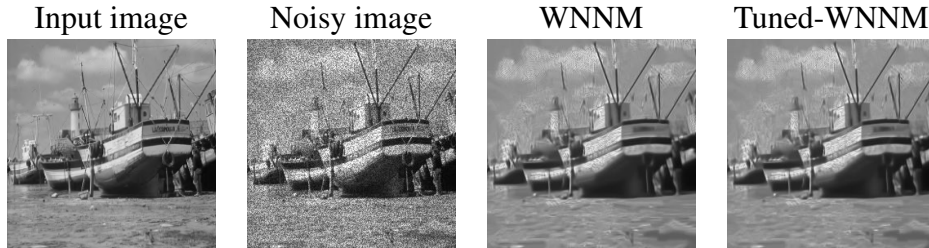


Figure 5.3: Input (256×256), noisy (speckle noise intensity $\sigma = 0.05$), and denoised images with WNNM and the tuned-WNNM.

better performance with low noise intensity, while the results improve when the speckle noise is higher. For example, WNNM has an average PSNR value of 22.32 with a speckle noise of intensity $\sigma = 0.3$, while tuned-WNNM has a PSNR value of 22.61.

5.2 Comparison among denoising methods on 2D videos and 3D images

We discuss the results of denoising arbitrary signals on synthetic signals (i.e., video and volumetric image) concerning the methods introduced in Sect. 4.2.1. For video denoising, we use generic 2D image sequences. For volumetric denoising, we consider a phantom through 3D spheres of different size, with faded boundaries. For both videos and volumetric data, artificial speckle noise with different intensities is applied to compare ground truth with denoised results.

Volumetric image Fig. 5.4 shows the denoising results of WNNM, BM4D, and Vidosat, with synthetic volumetric data with artificial speckle noise applied. WNNM has good results when we divide the volume in the same direction the filter has been applied, i.e., slice direction (Fig. 5.4(a)). In contrast, a scattering effect due to phase shift between adjacent slices is visible in the other directions, e.g., the white sphere in Fig. 5.4(b). This denoising approach does not consider any correlation between adjacent slices, generating artefacts. BM4D and Vidosat

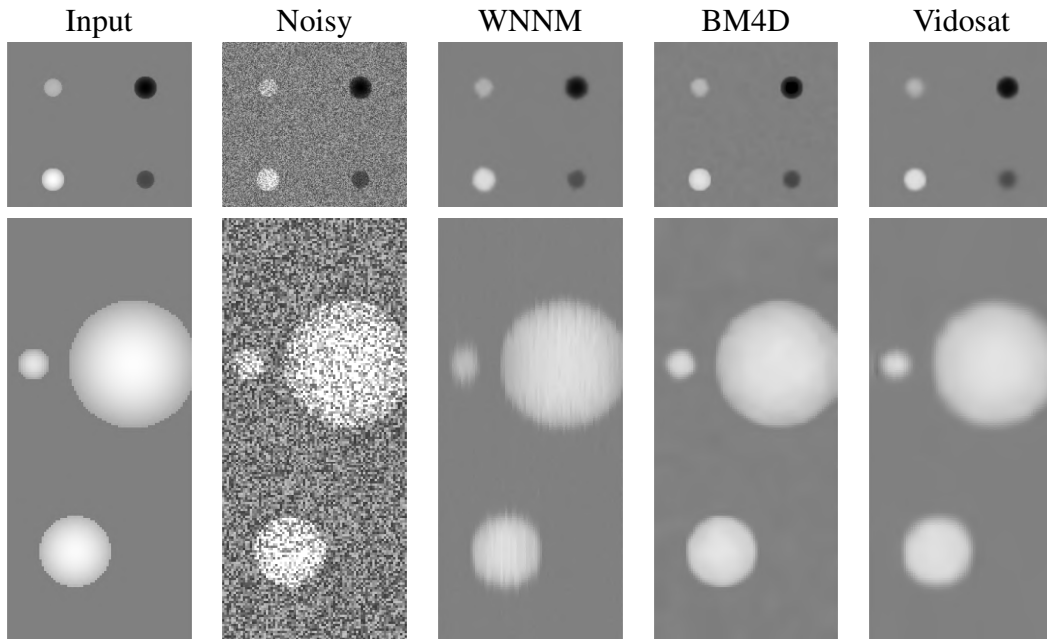


Figure 5.4: Denoising results of a phantom volumetric image with speckle noise applied: $230 \times 256 \times 80$ resolution, $\sigma = 0.05$ speckle noise.

have better results, both in terms of denoising and consistency among slices, in all the sections. In particular, BM4D better preserves edges, while Vidosat has a higher blurring effect. At the same time, Vidosat better removes noise than BM4D.

2D video Fig. 5.5 shows the denoising results of WNNM, BM4D, and Vidosat, with a synthetic 2D video with artificial speckle noise applied. BM4D and Vidosat show good results in terms of noise removal, and main features preservation (e.g., man details, books in the background); in particular, BM4D better reduces the blurring effect than Vidosat (e.g., man’s hand). WNNM shows a higher blurring effect, with a higher loss of details than the other two methods.

Quantitative comparison Synthetic data allow us to compute quantitative metrics between ground truth and denoised images. Table 5.3 summarise PSNR and SSIM values of the three denoising methods with volumetric data. Both BM4D and Vidosat have good results, while WNNM has slightly lower PSNR and SSIM values. Also, we underline that the results degrade when we increase the noise intensity. Table 5.4 reports the PSNR and SSIM values of the three denoising methods with 2D videos. Vidosat has the best results, as expected on 2D videos; however, BM4D has comparable results, showing the versatility of this method with different data structures.

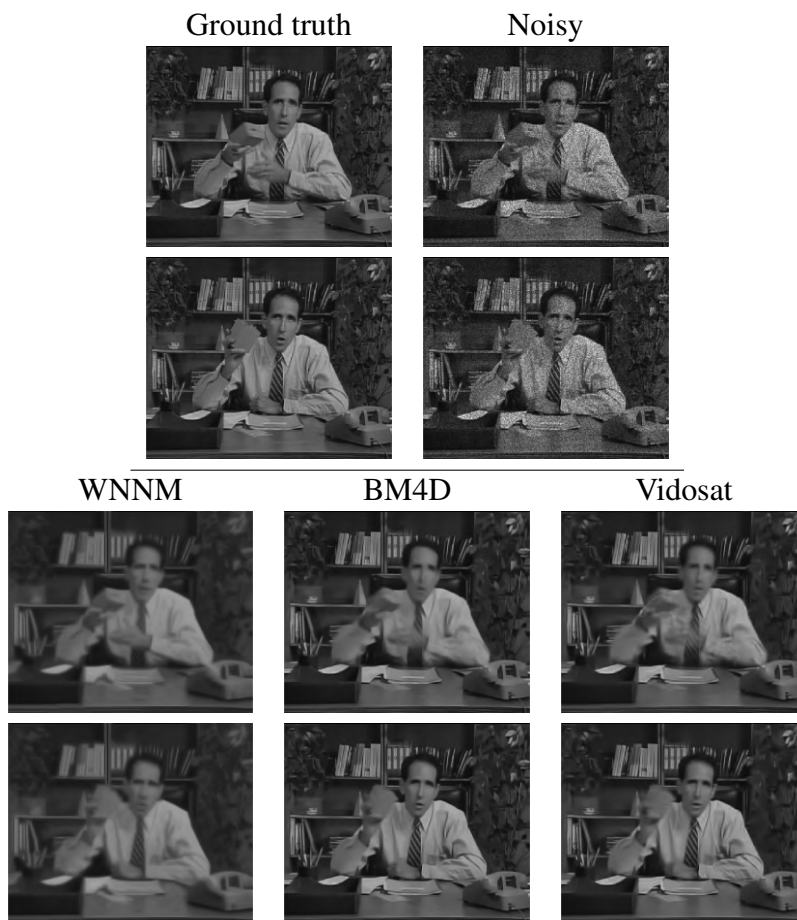


Figure 5.5: Denoising results on a 2D video with speckle noise applied: $352 \times 288 \times 40$ resolution, $\sigma = 0.025$ speckle noise.

5.3 Properties and tests of learning-based low-rank denoising

Signals acquired by digital sensors are generally affected by different types of noise, such as speckle [Bur78] and exponential [SSGL07] noise on biomedical images, salt-and-pepper [AZA18], Gaussian [Rus03], and Poisson [TZ12] noise on images acquired through camera sensors. In this context, we discuss the validation of the results of our low-rank denoising method (Sect. 3.3) on synthetic images with different types and intensities of noise.

Table 5.3: PSNR and SSIM metrics of the denoising methods tested on the volumetric data. Results refer to Fig. 5.4.

| Noise | Method | WNNM | BM4D | Vidosat |
|-------|--------|-------|--------------|--------------|
| 0.05 | PSNR | 39.18 | 39.95 | 39.45 |
| | SSIM | 0.935 | 0.983 | 0.988 |
| 0.1 | PSNR | 36.88 | 37.60 | 37.45 |
| | SSIM | 0.925 | 0.973 | 0.974 |
| 0.2 | PSNR | 33.32 | 34.5 | 35.12 |
| | SSIM | 0.923 | 0.956 | 0.985 |

Table 5.4: PSNR and SSIM metrics of the denoising methods tested on 2D video. Results refer to Fig. 5.5.

| Noise | Method | WNNM | BM4D | Vidosat |
|-------|--------|-------|-------|--------------|
| 0.05 | PSNR | 25.34 | 28.43 | 29.25 |
| | SSIM | 0.696 | 0.777 | 0.803 |
| 0.1 | PSNR | 25.01 | 28.16 | 28.92 |
| | SSIM | 0.636 | 0.780 | 0.798 |
| 0.2 | PSNR | 24.02 | 27.69 | 28.61 |
| | SSIM | 0.618 | 0.749 | 0.794 |

5.3.1 Denoising with optimal threshold values

The denoising with optimal thresholds (Sect. 3.3.1) is validated on synthetic images, comparing the ground truth with denoised images. Considering the Gaussian, speckle, and salt & pepper noise (Fig. 5.6), the denoised images after 20 iterations are very close to the input image, thus showing that the weighted nuclear norm minimisation recovers the ground truth image with any type of noise, when the proper thresholds are selected. Most of the details are well reconstructed, such as the cameraman’s face, the boat profile, or the mandrill zygoma. Images with a different noise, in terms both of type and intensity, have a different reconstruction accuracy. This result motivates us to distinguish learning-based networks according to the type of noise (Sect. 3.3.2). The optimal PSNR plot (Fig. 5.7, 1st row, blue line) provides the upper bound of the optimal denoising and allows us to compare the accuracy of our learning-based method with the optimal achievable results.

In Fig. 5.7(2nd row), the averaged optimal thresholds at the first iteration: $\bar{\lambda}_h^{it=1} = \sum_{j=1}^B \lambda_{hj}^{it=1} / B$, where B is the total number of 3D blocks. Different types of noise have similar behaviour; for example, in (a) the first singular value is reduced to an average value of 0.14, which is less than the 1% of the average input value. Then, the optimal thresholds increase up to 1.2 for the fourth singular value, which is more than the 73% of the input singular value. Finally, the thresholds decrease: the reduction of the 49th singular value is about 0.2.

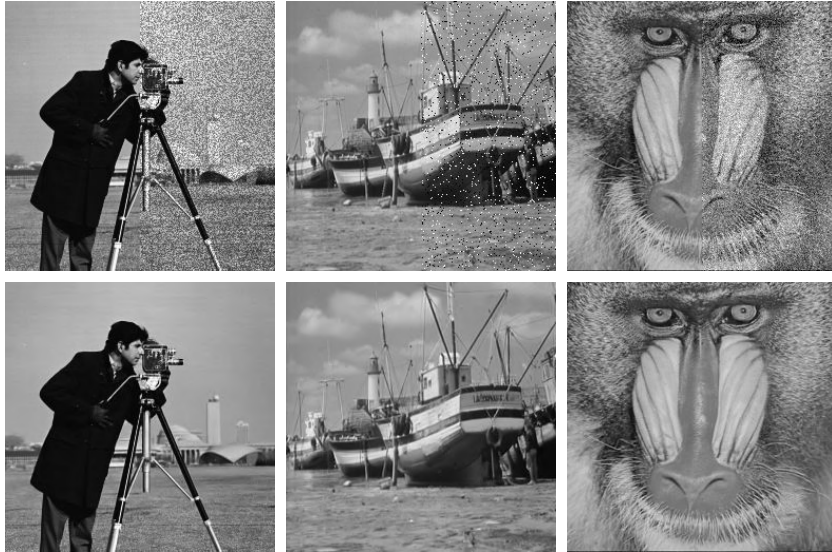


Figure 5.6: Input/noisy (1st row) and optimal denoising (2nd row) for: cameraman (256×256), with speckle noise and $\sigma = 0.05$; boat (256×256), salt & pepper noise with density 0.05; mandrill (256×256) Gaussian noise with $\mu = 0, \sigma = 0.01$.

Table 5.5: PSNR and SSIM metrics of our denoising, compared with previous work: (a) speckle, (b) salt & pepper, and (c) Gaussian noise.

| <i>Metrics</i> | <i>PSNR</i> | | | <i>SSIM</i> | | |
|----------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | (a) | (b) | (c) | (a) | (b) | (c) |
| Image Fig. 5.8 | (a) | (b) | (c) | (a) | (b) | (c) |
| BM3D | 28.05 | 26.20 | 25.99 | 0.786 | 0.720 | 0.734 |
| WNNM | 26.91 | 24.17 | 26.13 | 0.796 | 0.642 | 0.743 |
| NCSR | 27.69 | 25.64 | 25.89 | 0.794 | 0.689 | 0.727 |
| Our method | 28.69 | 27.02 | 26.01 | 0.843 | 0.758 | 0.729 |

5.3.2 Training and denoising

We present the learning-model training, the threshold values prediction, and the denoising results with different noise types and intensities, comparing our results with previous work.

Training results We select the SIPI data set composed of 44 images of different resolutions and subjects. We apply an artificial noise to the images (e.g., Gaussian noise with mean 0 and variance 0.01); then, we compute the 3D blocks with block-matching. We compute the input singular values and the target thresholds (Sect. 3.3.1). Images are classified according to the noise type (e.g., speckle, Gaussian) and intensity (e.g., $\sigma = 0.05$ in speckle noise) of the input image. Then, we split the data into $c = 5$ 3D block noise intensity classes, and $k = 3$ k -

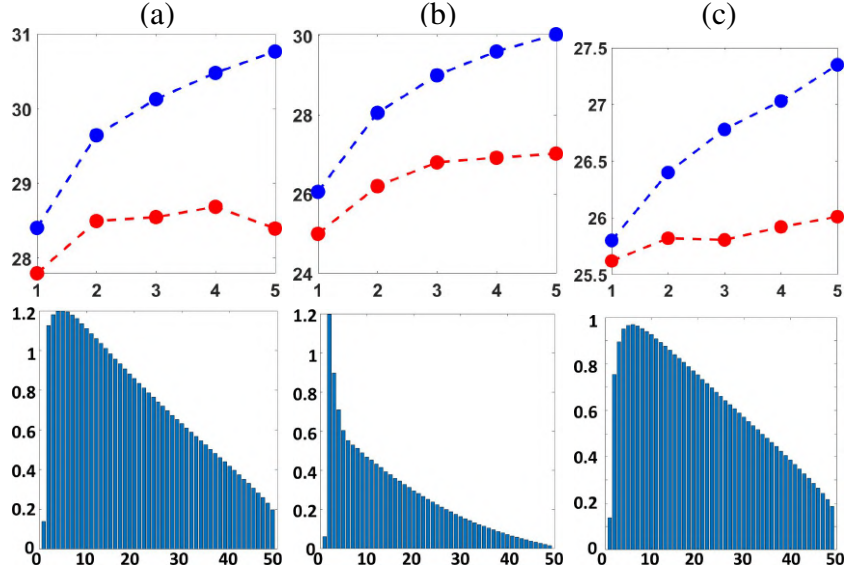


Figure 5.7: 1st Row: PSNR (y -axis) comparison of the number of iterations (x -axis) between optimal (Sect. 3.3.1) values (blue), and our method’s values (red). 2nd Row: average optimal thresholds (y -axis) of each singular value (x -axis). (a) Cameraman (256×256), speckle noise ($\sigma = 0.05$); (b): boat (256×256), salt & pepper noise (density 0.05); (c): mandrill, Gaussian noise ($\mu = 0, \sigma = 0.01$). Results refer to Fig. 5.8.

means labels (Sect. 3.3.2), for a total of 15 instances for each iteration. The final data set is composed of a set of 15 matrices $n \times N$, where N is the total number of 3D blocks, and n is the number of singular values (i.e., 49 in our tests, corresponding to a 7×7 patch). This approach is repeated at each iteration: for the computation of the data set at iteration t , the block matching is applied for $(t - 1)$ iterations. At each iteration, the proper networks previously computed by the minimisation model (Sect. 3.3.2) are used to predict the thresholds and denoise the related 3D block. At iteration t , the optimal thresholds are computed through an optimisation step (Eq. (3.1)). In our tests, we apply 5 iterations of the training, for a total number of 75 networks (i.e., $5 \times 3 \times 5$) for each type of noise and noise intensity. As a matter of example, the training data set of the Gaussian noise with $\mu = 0$ and $\sigma^2 = 0.01$ is composed of more than 257K elements, each one composed of a couple of vectors (i.e., input and optimal target) in \mathbb{R}^{49} . Fig. 5.10a shows the partition of these elements, according to the 3D block noise intensity and the k -means classification. Since each 3D block noise intensity class contains the same number of elements, we have a regular partition of the training data set. In contrast, the k -means classification does not provide a uniform partition of the elements; indeed, some networks are trained with more data than others. However, each network has a sufficient number of elements due to the huge data set; in fact, the minimum number of elements is $N = 4204$, related to 3D block noise class $c = 1$ and k -means label $k = 3$.

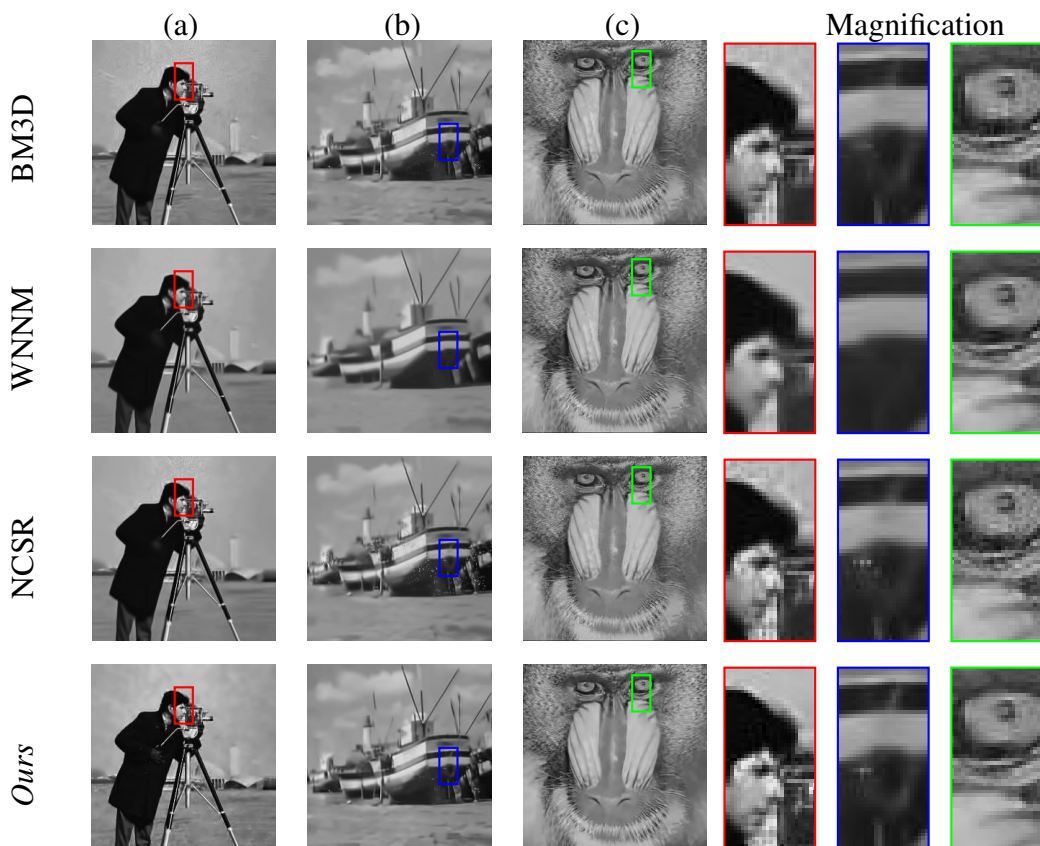


Figure 5.8: Comparison between previous work and our denoising: (a) cameraman (256×256), speckle noise with $\sigma = 0.05$; (b) boat (256×256), salt & pepper noise with density 0.05; (c) mandrill, Gaussian noise with $\mu = 0, \sigma = 0.01$; (d) magnification.

Computation time The computation of the input/optimal thresholds (Eq. (3.1)) and of the network weights (Sect. 3.3.2) are performed offline. For the computation of the training data set, we used 44 images, for a total number of 257,240 couples of input/target vectors for each noise type and iteration. The execution time for the generation of the data set of each noise type/intensity is approximately 120 minutes for each iteration. For the computation of the weights, the variables of each network are 49×49 matrices; we consider 3 types of noise (i.e. salt & pepper, speckle, Gaussian), 5 iterations for each noise, 5 3D block's noise intensity classes, and three $k - means$ labels, for a total number of 225 networks. The execution time of the training of each network is about 15 minutes. Tests have been executed with Matlab R2020a, on a workstation with 2 Intel i9-9900KF CPUs (3.60GHz), and 32GB RAM.

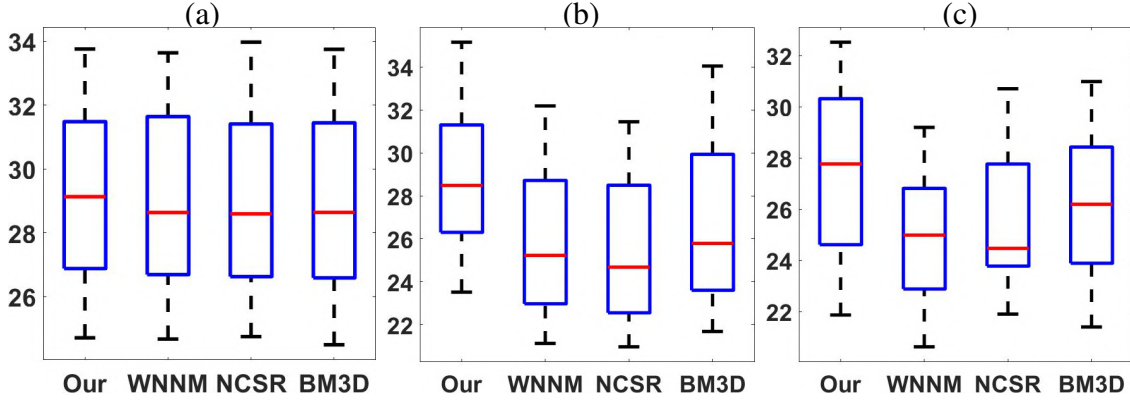


Figure 5.9: PSNR (y -axis) average values of our and state-of-the-art methods (x -axis) with Gaussian (a), speckle (b), salt and pepper (c) noise.

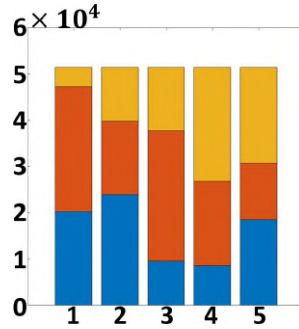


Figure 5.10: Partition of the elements (i.e., input and target vectors) for the training. Each noise intensity class (x -axis) is divided into 3 k -means classes (blue, orange, and yellow); y -axis shows the cumulated number of elements of each class.

Thresholds prediction Fig. 5.11b shows a comparison among the input, optimal, and predicted singular values of a specific 3D block. The optimal singular values reflect the thresholds in Fig. 5.7(2nd col.); for example, the shrinkage of the second singular value is larger than the first one. The predicted singular values are very close to the optimal singular values; in fact, the error for the prediction of a 49-dimensional vector is 0.6, which means an average error of 0.01 for each element of the vector. Analysing the error between the prediction and the optimal singular values $(\mathbf{Q}_{jk} - \hat{\mathbf{Q}}_{jk})_{j=1}^{49}$ (Sect. 3.3.2), the first values show a higher prediction error than the last ones (Fig. 5.11c), due to the different magnitude and variability of these values. However, there is not a clear correlation between these factors. The correlation between the prediction error of the thresholds and the data clustering (Sect. 3.3.2) shows that the error increases with the 3D block noise intensity (Fig. 5.11d), as the noise intensity implies a more irregular distribution of the singular values, which is harder to train and predict. Instead, the SVM-predicted classes and the prediction error of the network do not show any correlation.

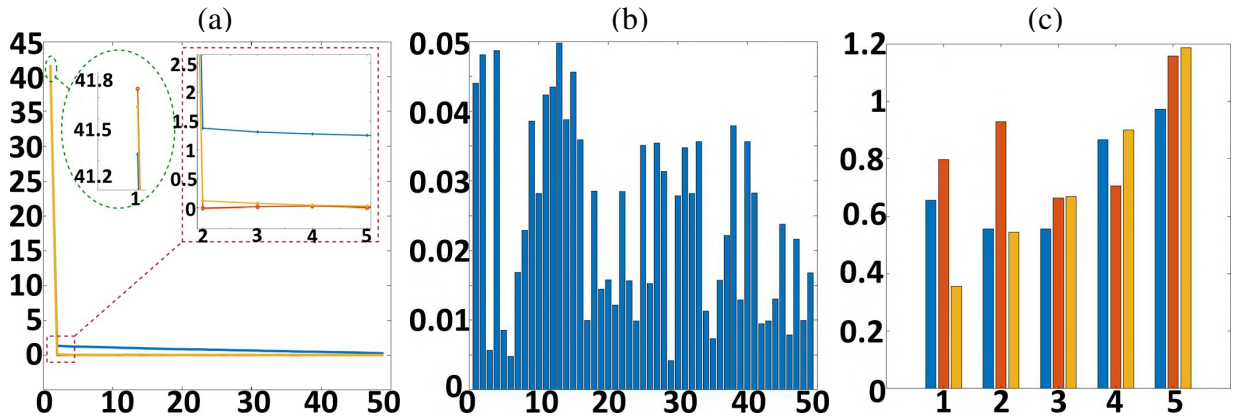


Figure 5.11: (a) Input (blue), optimal (orange), and predicted (yellow) singular values (y -axis) of the cameraman image with speckle noise, for the 49 (x -axis) singular values. (b) Prediction error (y -axis) of the 49 singular values (x -axis) of a 3D block of the cameraman image with speckle noise. (c) Partition of the prediction error (y -axis), when varying the noise intensity classes (x -axis): each class is composed of three SVM-predicted classes (i.e., blue, orange, and yellow).

Denoising results We discuss the generality of our method from the type of noise applied. Applying a speckle noise with $\sigma = 0.05$ (Fig. 5.8a), our method improves previous work and shows better preservation of the main features and details (e.g., the face of the man, the folds of the man’s trousers), and good noise removal (e.g., the sky in the background). Furthermore, our method does not introduce blurred patterns. In comparison, WNNM removes a larger amount of noise, while losing several details (e.g., man’s mouth), and increasing the blurring effect. On the boat image with salt & pepper noise with 0.05 density (Fig. 5.8b), our method better preserves the main details, such as the clouds, or the geometries of the boat. In comparison, BM3D does not remove the “salt” noise on the bottom side of the boat. On the mandrill image with $\mu = 0$ and $\sigma^2 = 0.01$ Gaussian noise (Fig. 5.8c), all methods have similar results, correctly removing the noise and preserving the main features of the image (e.g., the nose and the eyes of the mandrill). Only the mandrill’s pupil in the NCSR method is less uniform than our method.

Table 5.5 summarises the quantitative metrics of our method and previous work. With Gaussian noise, our method has comparable results with state-of-the-art methods, which are specific for this type of noise. In (c), the PSNR value is 26.01 for our method, 26.13 for WNNM (best result), and lower than 26 for BM3D and NCSR. Instead, our method has better results when other types of noise (e.g., salt and pepper, speckle) are applied. In Fig. 5.7, we compare the PSNR of our method with the optimal PSNR results (Sect. 3.3.1); in (a), our method has a PSNR value of 28.69 after 4 iterations, while the optimal PSNR value is 30.53. The error of the network prediction of the thresholds, even if small, leads our method to diverge from the optimal reconstruction. After five/six iterations, our method is not able to further improve the image

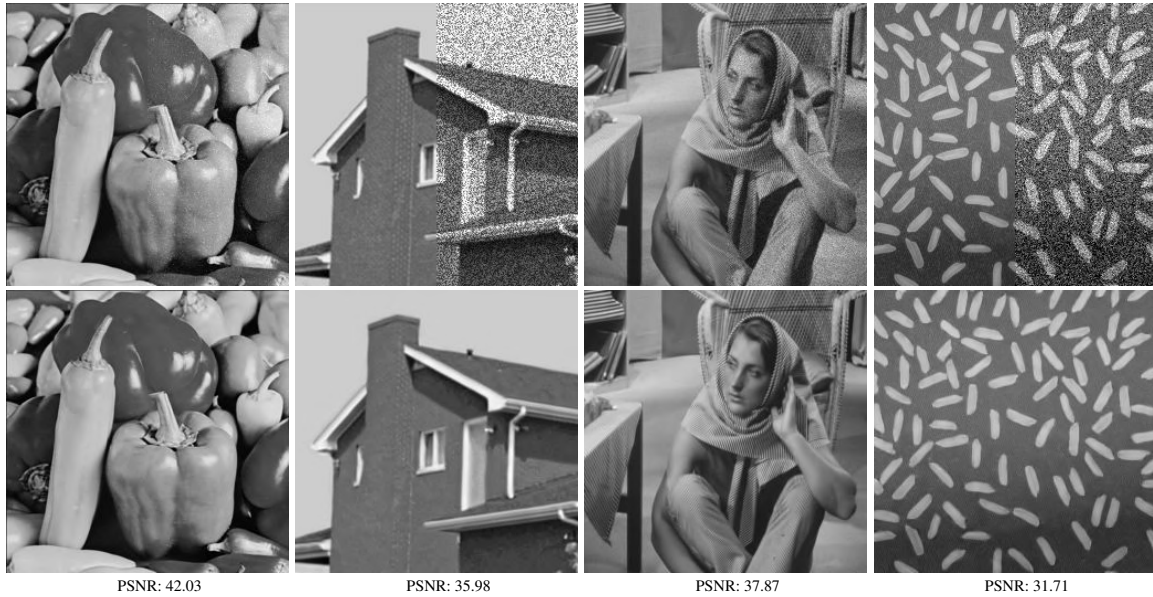


Figure 5.12: Input/noisy (1st row) and optimal denoising (2nd row) and respective PSNR for: peppers, exponential noise with $\mu = 0.02$; wall, speckle noise with $\sigma = 0.10$; Barbara, Poisson noise; rice, Gaussian noise with $\mu = 0, \sigma^2 = 0.02$.

approximation. Considering a large data set [RGBP18] of natural images of different resolutions and subjects, and applying different types of artificial noise, our method has slightly better results than previous work in the case of Gaussian noise, while it widely improves the PSNR values in the case of speckle and salt and pepper noise (Fig. 5.9).

Denoising results: additional noise types We further compare our method with previous work on four images with different noise types and intensity: Gaussian ($\mu = 0, \sigma^2 = 0.02$), speckle ($\sigma = 0.10$), exponential ($\mu = 0.02$), and Poisson (Fig. 5.12, first row). The denoising with optimal thresholds (Eq. (3.1)) has very high accuracy (Fig. 5.12, second row), showing that the weighted nuclear norm minimisation with the optimal thresholds recovers the ground truth image with any noise type and intensity.

Then, we show the generality from the noise type and intensity of our denoising method with predicted thresholds. In addition to previous work tested in Sect. 3.3.3, we compare our method with deep universal blind image denoising (DUBD) [SC21], shearlab [KLR16], and texture variation adaptive image denoising with nonlocal PCA (ACVA) [ZLLQ19]. Fig. 5.13 shows the average PSNR on a large data set [RGBP18]; our method outperforms previous work with all types of noise. In particular, our method has an average PSNR value of 38.02 with exponential noise, while the other methods have an average PSNR value lower than 33.00.

The visual comparison (Fig. 5.14) and the respective quantitative metrics (Table 5.6) show that

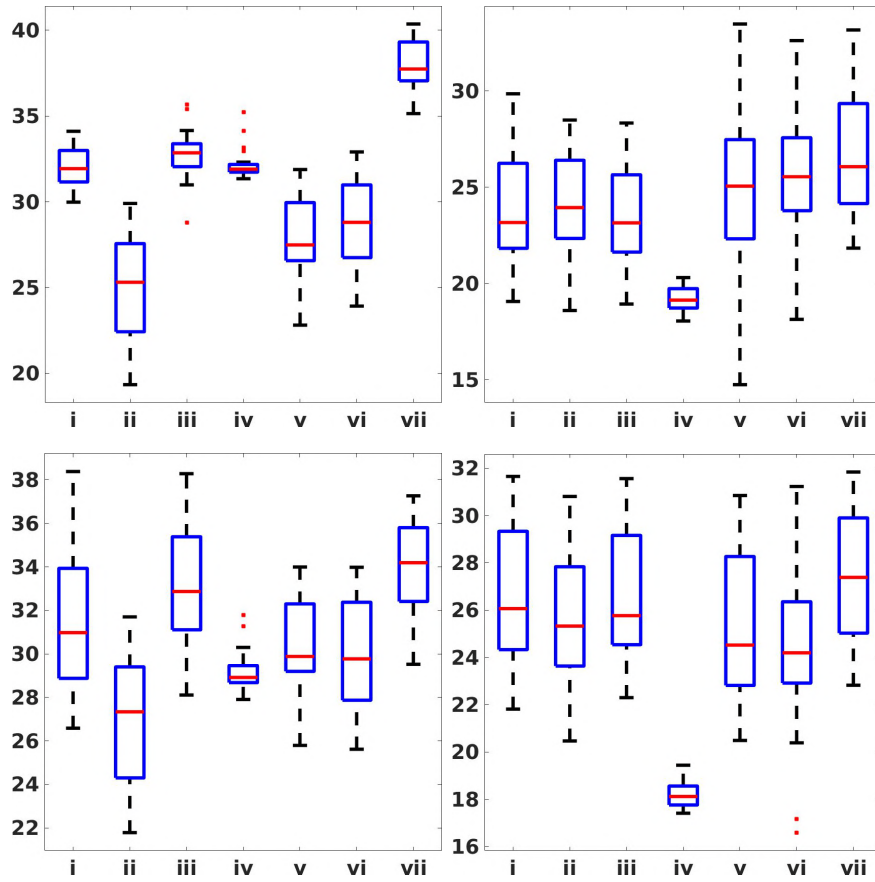


Figure 5.13: PSNR (y -axis) average values of our and state-of-the-art methods (x -axis) with exponential $\mu = 0.02$ (top-left), speckle $\sigma = 0.10$ (top-right), Poisson (bottom-left), Gaussian $\mu = 0$, $\sigma^2 = 0.02$ (bottom-right) noise. (i) BM3D, (ii) SHEARLAB, (iii) ACVA, (iv) DUBD, (v) WNNM, (vi) NCSR, (vii) Our denoising.

our method generally has better performance than all the other methods, in terms of edges preservation, noise removal, and low generation of artefacts. Learning-based method (DUBD) does not correctly remove noise in images where high-intensity noise is applied, such as speckle noise (b) and Gaussian noise (d). Shearlet-based method (SHEARLAB) generates artefacts with all types of noise, e.g., in the background of rice image (d); the preservation of edges and geometries is good. ACVA has good results with all types of noise, showing good generality properties. In particular, this algorithm correctly removes noise in (a,c,d), while it is not able to correctly process high-intensity speckle noise (b). BM3D does not correctly process high-intensity speckle noise (b). WNNM and NCSR show good results in terms of noise removal with all types of noise while losing some details (e.g., the edges of the window (b)).

Our method has better results than the aforementioned algorithms; it correctly processes different

Table 5.6: With reference to Fig. 5.14, we report the PSNR metric of our denoising, compared with previous work: (a) exponential, $\mu = 0.02$, (b) speckle, $\sigma = 0.10$, (c) Poisson, and (d) Gaussian noise, $\mu = 0$, $\sigma^2 = 0.02$.

| PSNR | (a) | (b) | (c) | (d) |
|------------|--------------|--------------|--------------|--------------|
| DUBD | 31.79 | 16.01 | 28.11 | 17.35 |
| SHEARLAB | 23.40 | 27.89 | 26.85 | 25.82 |
| ACVA | 32.95 | 25.25 | 34.11 | 28.12 |
| BM3D | 32.18 | 25.56 | 33.98 | 27.98 |
| WNNM | 32.08 | 28.29 | 33.87 | 28.45 |
| NCSR | 32.20 | 28.37 | 33.30 | 27.15 |
| Our method | 38.82 | 29.18 | 34.43 | 28.18 |

types of noise (a,c,d), preserving the geometries (e.g., the edges of peppers and rice grains), without generating artefacts. Furthermore, our method well denoises high-intensity speckle noise (b), removing noise while preserving main geometries.

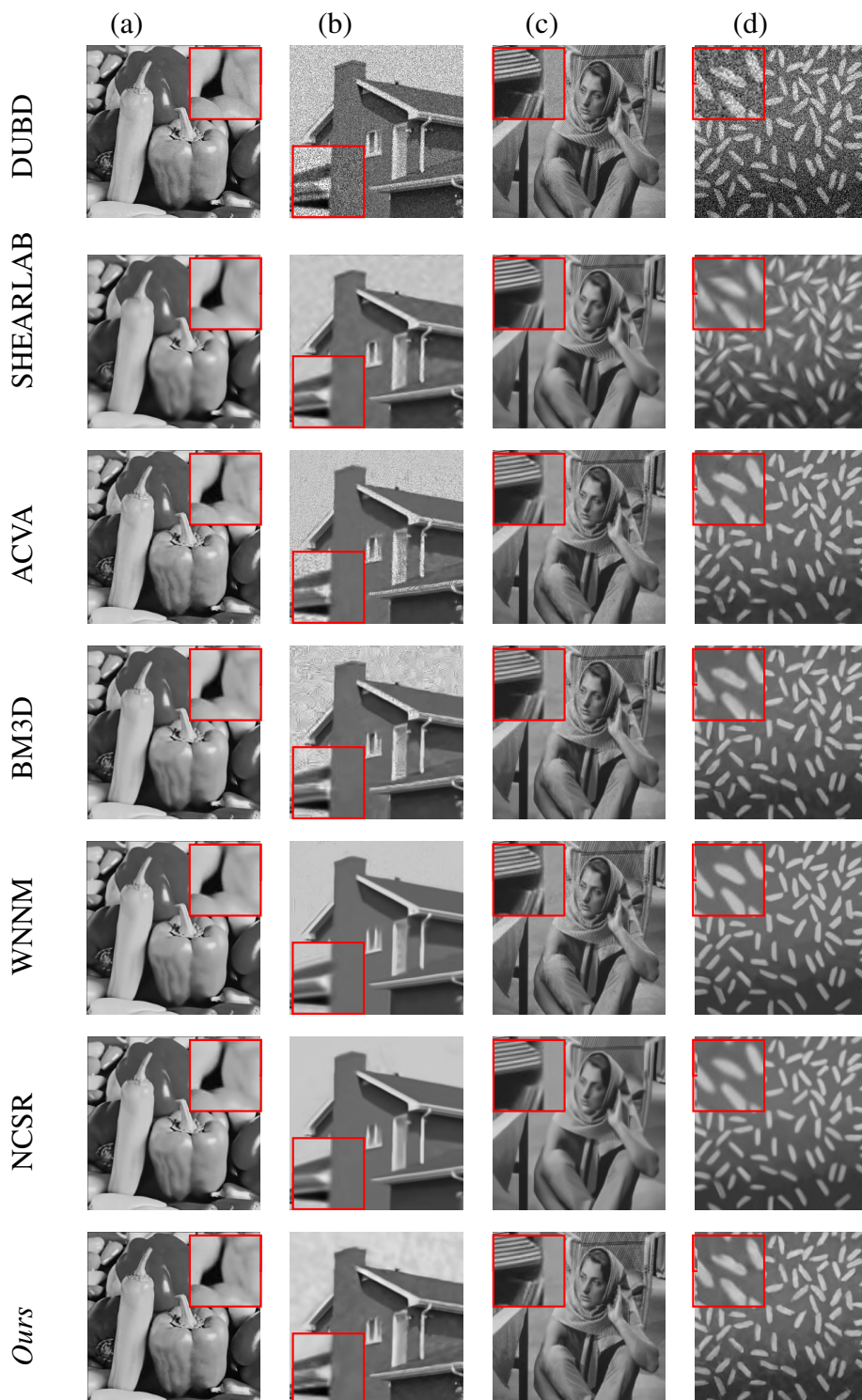


Figure 5.14: Comparison between previous work and our denoising: (a) peppers, exponential noise with $\mu = 0.02$; (b) wall, speckle noise with $\sigma = 0.10$; (c) Barbara, Poisson noise; (d) rice, Gaussian noise with $\mu = 0, \sigma^2 = 0.02$.

PART III

**Real-time US super-resolution:
contributions and novelties**

Chapter 6

Real-time super-resolution of US 2D images and videos

Ultrasound acquisition has many advantages compared to magnetic resonance and tomography, such as its efficiency, cheapness, and non-invasiveness. Furthermore, its real-time acquisition provides instantaneous feedback to the physician, for example during regional anaesthesia. Through US videos, the physician analyses the temporal variation of an anatomical feature (e.g., the movement of a muscle, the volume of the ventricle), which can be generated either by the shift of the probe or by the movement of the anatomical part. 2D US videos are acquired through 2D probes, which capture sequences of images at a given frequency. The resolution of each image is affected by the required frequency of the video, since some anatomical districts (e.g., cardiac) require a high acquisition frequency, to accurately acquire the behaviour of anatomical features that quickly change over time.

Our goal is the design of a novel deep learning framework for the super-resolution of 2D US images, by increasing the image resolution and reconstructing non-acquired scan lines. Applying our approach to US videos with a low spatial resolution and a high frequency (e.g., for the cardiac district), we can generate high-frequency 2D US video with an increased spatial resolution of each frame, thus overcoming the main limits of current US probes, whose spatial resolution decreases as the acquisition frequency increases.

First, we compare several state-of-the-art up-sampling algorithms among the ones discussed in the related work section (Sect. 2.2.2) and identify the best method in terms of quantitative metrics and visual evaluation. Then, we train a neural network to improve the results of the up-sampling, to match the target image (i.e., the high-resolution image). Our network does not perform the interpolation of the missing lines; in fact, this task is already performed by up-sampling. In contrast, our network learns how to transform the up-sampled lines into the target lines. To improve the quality of the up-sampling, we train multiple networks, each one specialised to the input

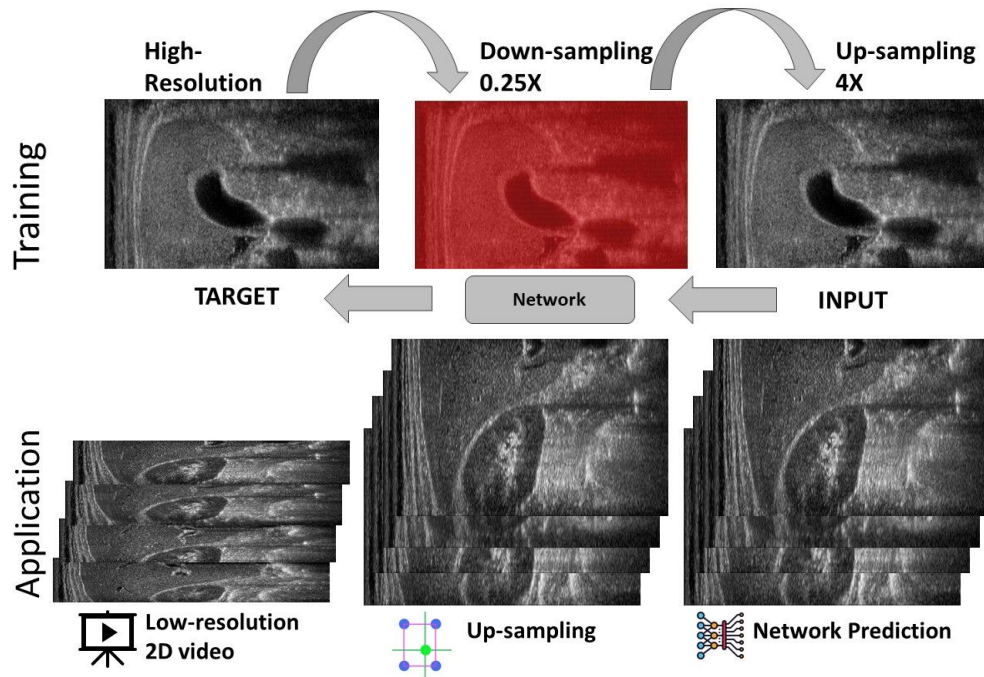


Figure 6.1: Proposed framework: training of the learning-based model and spatial up-sampling of US videos.

anatomical district (e.g., cardiac, abdominal) and its low-resolution image (e.g., 0.5X, 0.25X). This specialisation improves the quality of the up-sampling since we specialise the network to a specific prediction. The execution time of the super-resolution depends on the up-sampling and the network prediction; the prediction is achieved in real-time on standard medical hardware.

As the main contribution (Sect. 6.1), we propose a novel learning-based architecture, which improves the *Wide Activation for Efficient and Accurate Image Super-Resolution* (WDSR)[YFH20]. The kernel size is selected according to the dimension of the low-resolution image to guarantee that at least two original lines (i.e., two lines that are acquired by the probe) are always included in the convolution operation. Then, we modify the loss function to improve the visual accuracy of the prediction. Our logarithmic-based loss includes only up-sampled lines, excluding lines acquired by the probe.

Our approach is general in terms of the building blocks of the framework; in fact, we can select different up-sampling algorithms, e.g., *Single Image Super Resolution* (SISR) [PE14], *Enhanced Super Resolution Generative Adversarial Network* (ESRGAN) [WYW⁺18] and deep learning architectures, e.g., *Pix2Pix* [Izze17] and *VGG19* [SZ14]. As experimental validation (Sect. 6.2), we perform a quantitative and qualitative evaluation of our framework on a large collection of US images acquired from different anatomical (e.g., muscle-skeletal, obstetric, abdominal) districts.

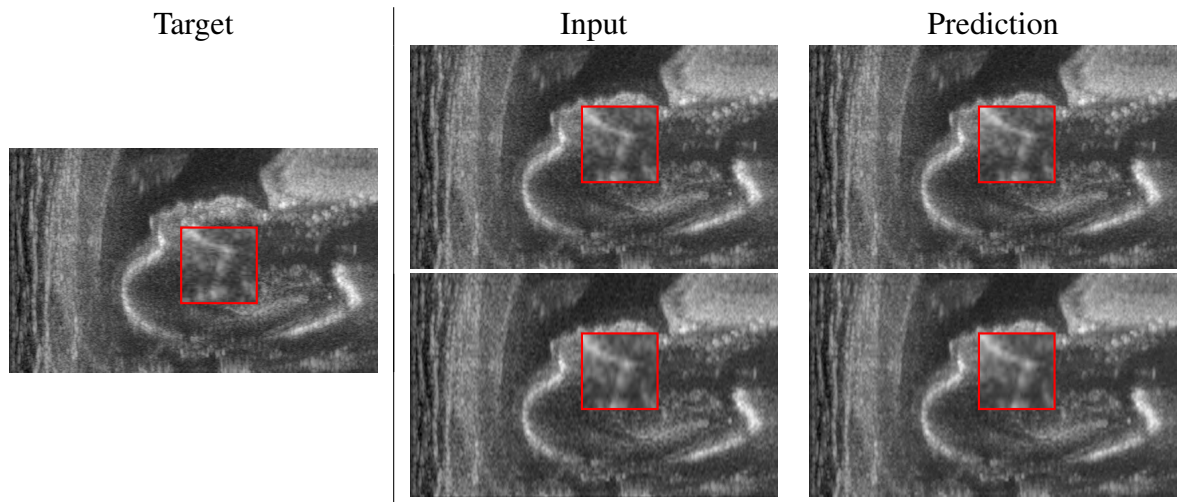


Figure 6.2: Prediction on the raw images of the obstetric district: 2X up-sampling (first line); 4X up-sampling (second line).

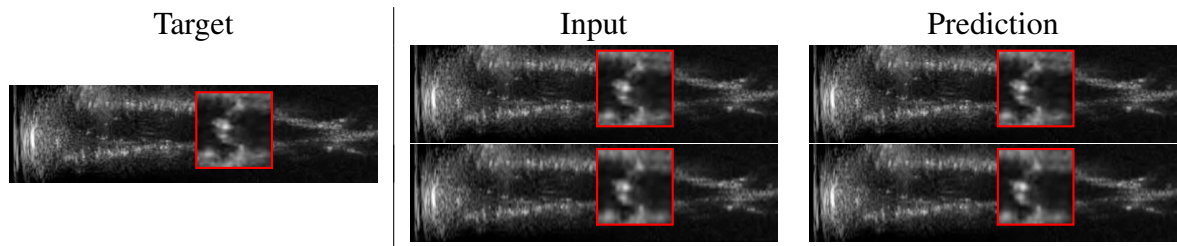


Figure 6.3: Prediction on the raw images of the cardiac district: 2X up-sampling (first line); 4X up-sampling (second line).

Related publications

Cammarasana, Simone, Paolo Nicolardi, and Giuseppe Patanè, “*Super-Resolution of 2D Ultrasound Videos.*”, submitted at Medical & Biological Engineering & Computing [Journal Paper]

6.1 Real-time super-resolution framework

Ultrasonic sound waves are emitted by the probe and straightly propagate till they encounter a tissue variation; the reflected echo depends on the tissue property (i.e., *acoustic impedance*), which is measured by the probe. The echo signals are processed and combined to generate the

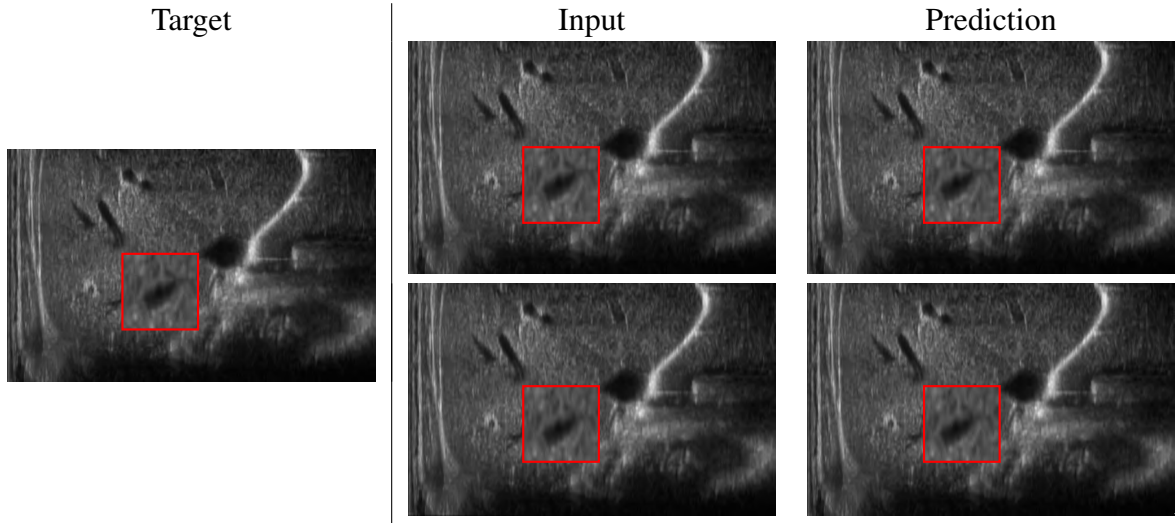


Figure 6.4: Prediction on the raw images of the abdominal district: 2X up-sampling (first line); 4X up-sampling (second line).

underlying image, which has a resolution of $l \times d$, where l is the number of scan lines (i.e., the lateral resolution), and d is the depth of the acquisition of each scan line (i.e., the axial resolution). The resolution of the image in terms of lateral direction (i.e., the direction perpendicular to the US propagation along the scan line) depends on the number of elements (i.e., the *piezoelectric crystals*), which are activated to generate the US waves. In this context, we focus on the super-resolution of US images to increase the lateral resolution of the image; this is relevant to improve the quality of the image, its visual interpretation by the physician, and post-processing steps, e.g., classification [ANMM⁺17], diagnosis [BSGR⁺21], and segmentation [LZT⁺22].

We also mention that our super-resolution framework can be extended to the up-sampling of ultrasound localisation microscopy imaging where a high acquisition frame rate is required [EPP⁺15]; this approach allows us to acquire anatomical structures in the order of tens of micrometers and reconstruct through our super-resolution scheme the missing information in the lateral direction.

Proposed framework Our framework is composed of two steps: first, we up-sample the low-resolution image through an interpolating method; after the comparison of state-of-the-art methods (Sect. 6.2.2), we select *Cubic Convolution* as the up-sampling algorithm. Then, we apply a learning-based network to improve the visual accuracy of the up-sampling.

We consider the *Esaote data set*, which contains more than 10K US images at different resolutions, and is acquired from different anatomical districts (e.g., obstetric, cardiac). Given a high-resolution image (i.e., the target) acquired by the probe, we build the corresponding low-

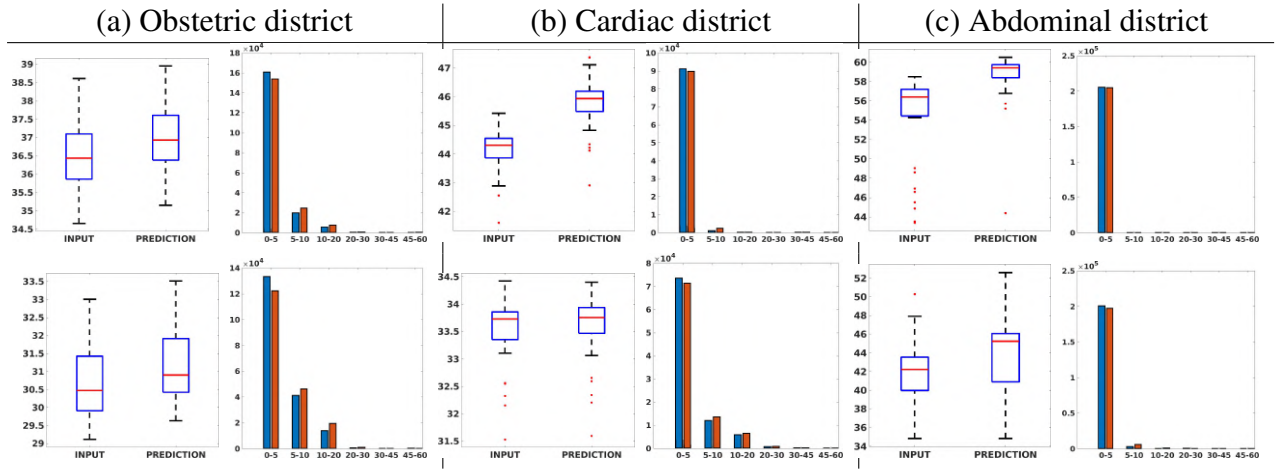


Figure 6.5: PSNR box-plot district (left) of the (a) obstetric, (b) cardiac, and (c) abdominal districts, and error histogram (right): prediction (blue) vs. input (red): 2X (first line) and 4X (second line) results.

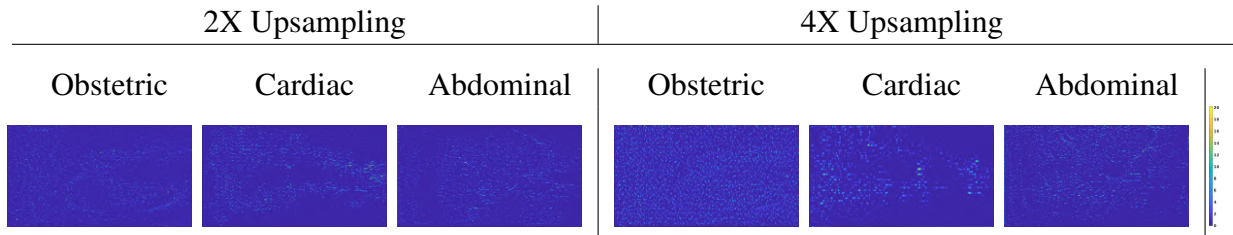


Figure 6.6: Concerning Figs. 6.2,6.3,6.4, we report the absolute value of the distance between the input and the prediction, for both 2X (first row) and 4X (second row) up-sampling factors.

resolution image by removing one line each 2 (0.5X) or 4 (0.25X). This approach is consistent with the acquisition of the US image, where the probe can acquire at the full, half, or a quarter of the maximum number of scan lines, depending on the activation of the piezoelectric crystals. We up-sample the low-resolution images through *Cubic Convolution* at 2X (applied to 0.5X low-resolution) or 4X (applied to 0.25X low-resolution). Then, we use the couples of up-sampled and target high-resolution images to analyse the proposed framework, through the training and the prediction of the learning-based network, with a specialisation in anatomic districts.

We generate a separate data set of 1.5K images for each anatomical district and two different up-sampling resolutions of 2X and 4X. Our approach requires the interpolation of the missing rows to the up-sampling method, while the learning model deals with the prediction of the target values from the interpolated values of the up-sampling method (Fig. 6.1). As *quantitative metrics*, we consider the PSNR for the comparison of the high-resolution target with both the up-sampled image and the prediction of the network. We also compare the histogram of the absolute value of the prediction error, to analyse the number of pixels whose error is lower than a certain threshold.

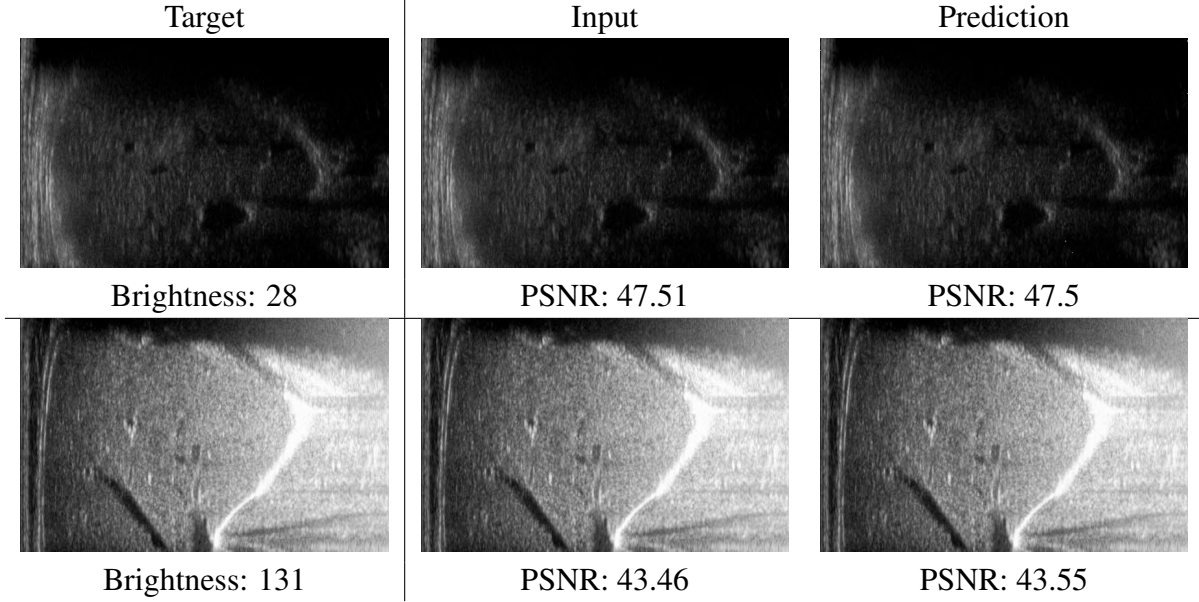


Figure 6.7: Input and prediction of the raw images of the abdominal district 2X with different levels of brightness: low brightness (first row) and high brightness (second row).

Deep learning network We select WDSR [YFH20], an architecture that exploits residual blocks since it improves the prediction of images where the difference between the input and the target is small. We propose an improved version of this network: *custom-WDSR*. Given an $y = L \times D$ target image, and its approximation \hat{y} , our loss function is defined as

$$Loss(y, \hat{y}) = \begin{cases} \sum_{l,d=1}^{L,D} \log \frac{|y_{ld} - \hat{y}_{ld}| + \epsilon}{k}, & \text{mod}(l, s) = 0, \\ 0, & \text{otherwise,} \end{cases}$$

where s allows us to exclude the lines acquired by the probe (e.g., $s = 4$ when 4X up-sampling is applied); $\epsilon = 10^{-4}$ avoids a null error for the logarithmic loss, and $k = 5$ determines the curvature of the logarithmic loss function. We enhance the pixels where the loss is less than 5 on the 0-255 range, to improve the visual similarity between the prediction and the target image. The size of the kernel of the convolution filter depends on the up-sampling factor; in the case of a 2X up-sampling, we apply a 3×3 filter; for a 4X up-sampling, we apply a 5×5 filter. This choice allows us to include at least two lines acquired by the probe, in the convolution operator. Finally, we set the number of layers to 16 and the number of kernels to 10. The learning rate iteratively decreases, up to 10^{-6} , and the number of epochs is set to 200. The input and output layers of the network are $\#batch \times L \times D$ size.

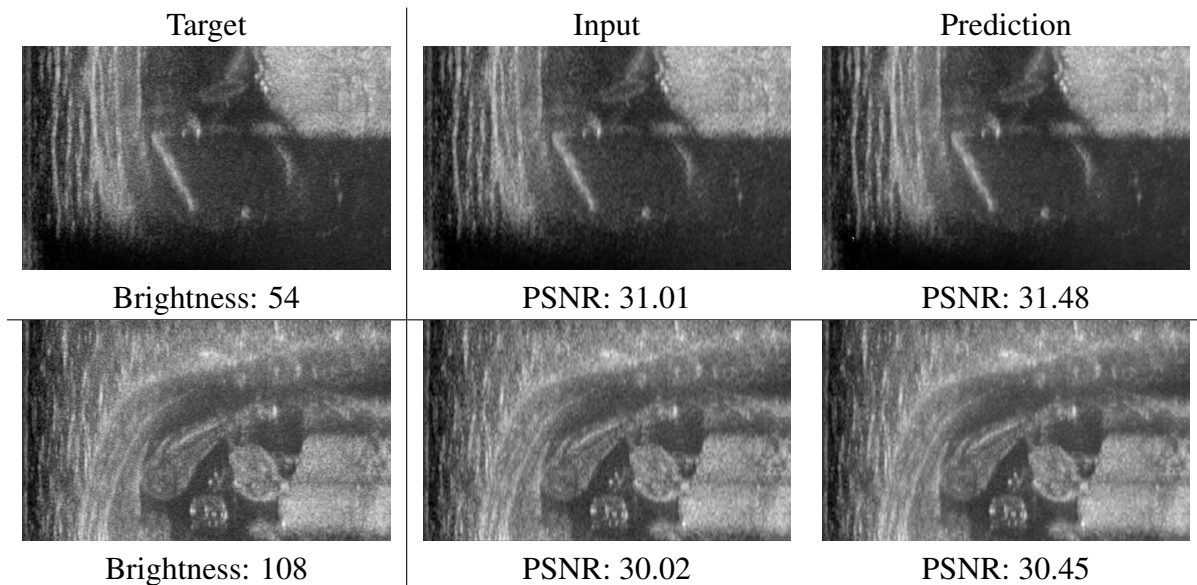


Figure 6.8: Input and prediction of the raw images of the obstetric district 4X with different levels of brightness: low brightness (first row) and high brightness (second row).

6.2 Experimental results on US 2D images and videos

We discuss the results of the proposed super-resolution of 2D US images and videos (Sect. 6.2.1) and compare our results with previous work (Sect. 6.2.2).

6.2.1 Framework prediction

Our real-time framework applies to US 2D images, with an image-to-image learning model. Then, the application to each frame of a sequence of frames allows us to increase the spatial resolution of a US 2D video.

Super-resolution of US 2D images We train each learning-based network (*custom-WDSR*) with 1.5K images, where the input is the outcome of the selected up-sampling method (i.e., *Cubic convolution*), and the target is the original high-resolution image. We recall that input and target images have the same resolution, as the reconstruction of the missing lines has been already performed by *Cubic convolution*. Figs. 6.2,6.3,6.4 show the results of the network prediction, compared with the input and the target images.

Our framework visually improves the results, in terms of blurring and artefacts. This result is more evident in the magnification of the ear of the foetus (Fig. 6.2), the mitral valve (Fig. 6.3),

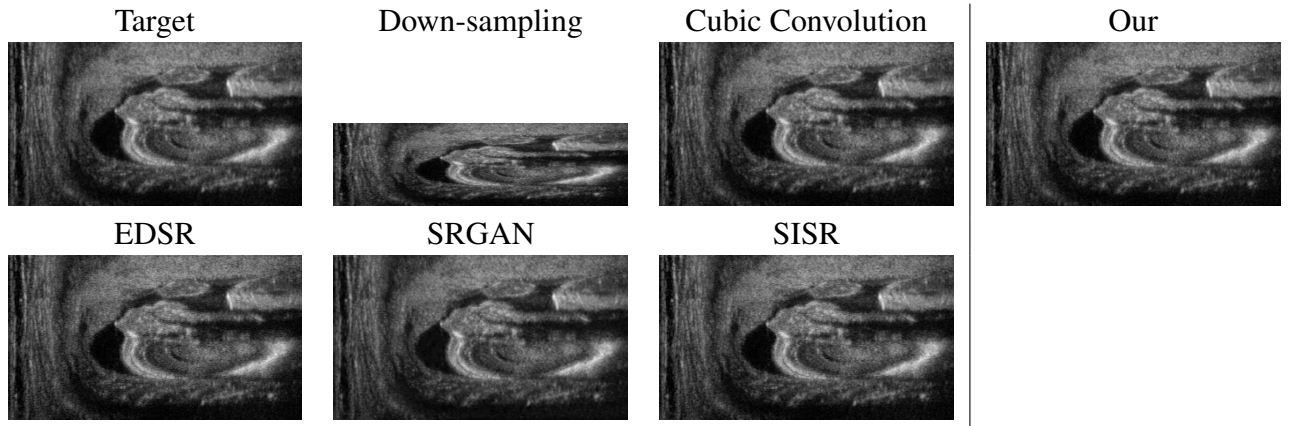


Figure 6.9: Comparison of up-sampling methods vs. our method on the obstetric district: 0.5X low-resolution and 2X up-sampling. See also Table 6.1.

and the mass edges (Fig. 6.4). Fig. 6.5(a-b-c, left) shows the box plot of the quantitative metrics, comparing the target images with the prediction and the *Cubic convolution*, respectively. The metrics are computed on a data set of 50 images of the same district and with the same up-sampling factor. We report that the PSNR median value improves of 1.5% on obstetric 4X images, 3.7% on cardiac 2X images, and 7.1% on abdominal 4X images.

Fig. 6.5(a-b-c, right) shows the histogram of the absolute value of the error with respect to the target image, of the prediction and *Cubic convolution* results, respectively. The histograms show the number of pixels where the prediction error is lower than 5 (i.e., the first bin of the histogram), which means very similar to the target when visually analysing the images. From the *Cubic convolution* to the predicted images, this value increases of 9.0% on obstetric 4X images, 2.9% on cardiac 4X images, and 2.0% on abdominal 4X images.

The analysis of the absolute value of the difference between the input and the prediction of the network (Fig. 6.6) shows that the alteration of our prediction to the pixel values ranges from 0 to a maximum absolute value of 20; furthermore, the black uniform areas are less affected by the prediction. In terms of the distance between the input and the prediction, we do not observe a significant difference among anatomical districts and between 2X and 4X up-sampling.

We also verify the robustness of our method on images at different brightness. Characterising the brightness of an image as the average value of all pixels, we test images with high and low brightness on different anatomical districts and up-sampling factors. Figs. 6.7, 6.8 show that the prediction performed with our trained network is robust to different value of image brightness, never lowering the output accuracy or generating artifacts. Comparing the input and the prediction of our network with the target image, we improve the PSNR value from 43.46 to 43.55 with high brightness images from the abdominal district 2X up-sampling, and from 31.01 to 31.48 with low brightness images from the obstetric district 4X up-sampling.

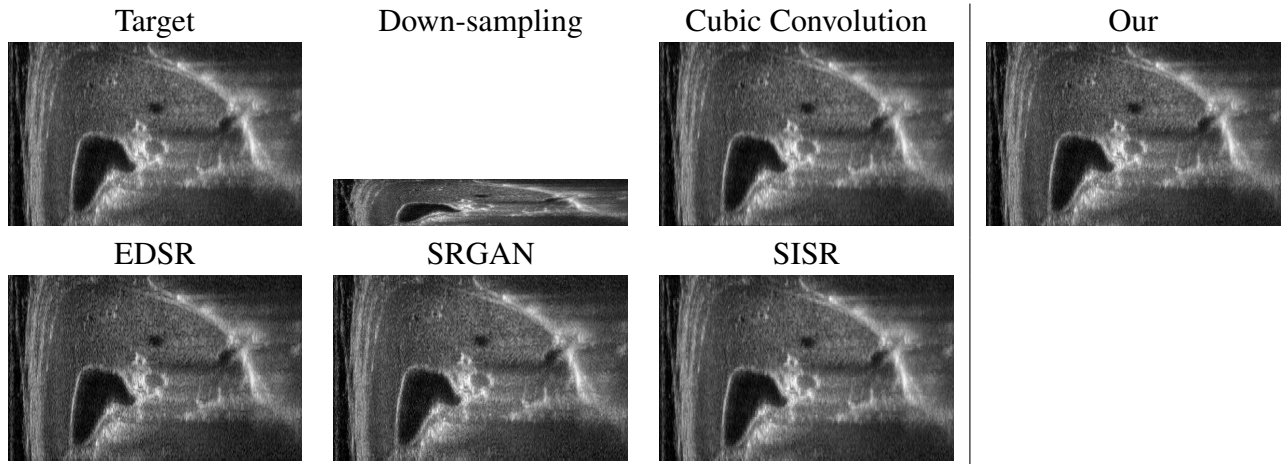


Figure 6.10: Comparison of up-sampling methods vs. our method on the abdominal district: 0.25X low-resolution and 4X up-sampling. See also Table 6.1.

Super-resolution of US 2D videos Applying our approach to US videos with a low spatial resolution and a high frequency (e.g., for the cardiac district), we can generate high-frequency 2D US video with an increased spatial resolution of each frame, thus overcoming the main limits of current US probes, whose spatial resolution decreases as the acquisition frequency increases. The relationship between image resolution and video frequency f is given by $f = c/(2 \cdot d \cdot l)$, where c is the speed of sound. The acquisition of low-resolution US images allows the physician to increase the acquisition frequency. The probe acquires a reduced number of lines: we refer to 0.5X and 0.25X low-resolution images, as $l/2 \times d$ and $l/4 \times d$ resolution, respectively. For the experimental tests on the spatial super-resolution of 2D US videos, we refer the reader to the uploaded video (see URL below). In the video, the input signal is a 2D US video at full resolution $L \times D \times T$ with L lines, D depth, and T frames. We down-sample each image at $L/2$ or $L/4$, and apply our framework for the spatial super-resolution, to reconstruct the full resolution 2D video. Video URL: <https://www.dropbox.com/s/p42pzxxvgf9gacl/SuperResolution-US.mp4?dl=0>.

Execution time and computational cost The experimental tests on the CINECA Cluster are supported by the ISCRA-C Scientific Project *US-ML-SR*. We define a HPC implementation of the proposed framework on the CINECA-Marconi100 cluster, exploiting both CPUs (IBM POWER9 AC922) and GPUs (NVIDIA Volta V100). We train multiple networks with large data sets for the target medical application, thus increasing the specialisation of anatomical districts, and consequently the accuracy of the prediction. The training data set can be periodically updated with the up-sampled images after the expert validation of the super-resolution results. To test the training phase of the learning-based networks in the HPC environment, we exploit 8 nodes, each one composed of 32 cores and 4 accelerators, for a theoretical computational performance

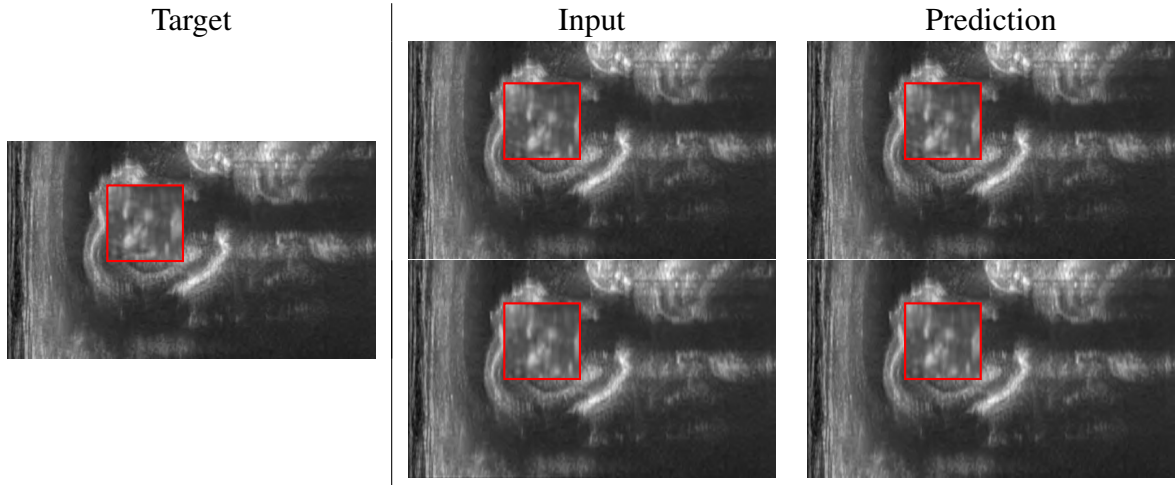


Figure 6.11: Prediction on the denoised images of the obstetric district: 2X up-sampling (first line); 4X up-sampling (second line).

of 260 TFLOPS, and 220 GB of memory per node.

The computational cost of the prediction depends on the resolution of the input image and on the architecture of the network: in particular, the computational cost of a convolution operation is $\mathcal{O}(r/s_r \cdot c/s_c) \cdot (f_r \cdot f_c) \cdot f$; in our application, the input images have variable resolutions, with a maximum value of $r = c = 600$, the kernel-filter size on rows and columns is $f_r = f_c = 3$ on 2X applications and $f_r = f_c = 5$ on 4X applications, the stride on rows and columns is $s_r = s_c = 1$, we use 16 convolution operators, and 10 kernel filters.

We test the prediction on GPU-based hardware, which replicates the hardware of a US scanner currently in use. Given a set of US input images from different districts at different resolutions, the average execution time is 14 milliseconds.

Super-resolution with denoised images To evaluate the effect of denoise pre-processing for the super-resolution of US images, we apply a soft denoising filter to the input raw images. In particular, we apply our low-rank denoising method 3.3, which allows us to select the intensity of the smoothing. This approach generates denoised images that are visually similar to raw images, and simultaneously more uniform. Then, we generate down-sampled images (0.5X and 0.25X) and apply the *Cubic convolution* up-sampling. These couples of images (i.e., denoised at full resolution and up-sampled) are used to train the learning-based network (Sect. 6.1). With this approach, we verify the performance of both the up-sampling algorithm and our learning-based prediction when applied to the input denoised images.

Fig. 6.11 shows the results of the prediction of the network, compared with the input and the

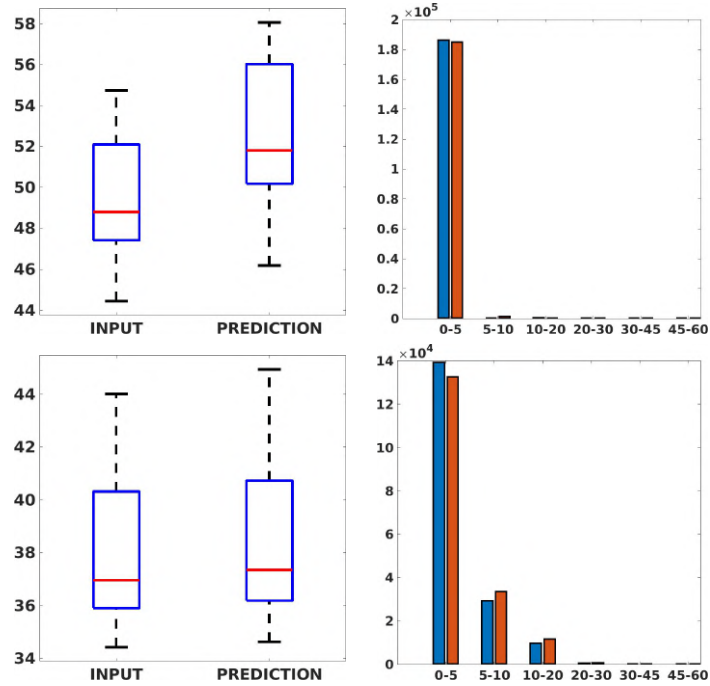


Figure 6.12: PSNR box-plot of the obstetric district with denoised images (left) and error histogram (right): prediction (blue) vs. input (red): 2X (first line) and 4X (second line) results.

target denoised images of the obstetric district. Our framework visually improves the results, in terms of blurring and artefacts. Fig. 6.12 (left) shows the box plot of the quantitative metrics, comparing the target images with the prediction and the *Cubic convolution*, respectively. The PSNR metric is computed on a data set of 50 images, belonging to the same district, and with the same up-sampling factor. Concerning the corresponding raw images (Fig. 6.5 (a, left)), the denoising allows the network to significantly improve the results of the up-sampling and the prediction. In particular, comparing the target images with the predicted images, the median PSNR value of obstetric 2X denoised images is 51.8, compared to the median PSNR value of obstetric 2X raw images which is 36.9.

Fig. 6.12 (right) shows the histogram of the absolute value of the error with respect to the target, of the prediction and *Cubic convolution* respectively. This result shows that our framework increases of 1.7% and 14% (2X and 4X, respectively) the number of pixels where the prediction error is lower than our defined threshold, and the prediction is very similar to the target when visually analysing the images, and improved compared to the learning framework applied to raw images. Finally, the denoise pre-processing can be performed in real-time through our learning-based denoising framework (Chapter 3).

Table 6.1: Concerning Figs. 6.9, 6.10 we report the PSNR metric computed between the target and each up-sampling method, as the mean value among the 50 test images.

| Test | Obstetric 2X | Abdominal 4X |
|-------------------|--------------|--------------|
| Cubic Convolution | 36.52 | 42.17 |
| EDSR | 32.08 | 34.91 |
| SRGAN | 33.70 | 36.35 |
| SISR | 34.75 | 38.58 |
| OURS | 37.00 | 44.35 |

6.2.2 Comparison with previous work

We show both the comparison among state-of-the-art algorithms that are used for the selection of the up-sampling method of our framework and the comparison of our results with previous work. Among up-sampling STAR methods, we test four methods belonging to different classes: *Cubic Convolution* [Key81], a kernel-based interpolating method; *Enhanced Deep Residual Networks - EDSR* [LSK⁺17], a learning-based method trained on generic images; *Enhanced Super-Resolution Generative Adversarial Network Plus - ESRGAN+* [RR20], a learning-based GAN method, specialised on US images with a dedicated training; *Single Image Super Resolution - SISR* [PE14], an up-sampling method which exploits sparse representations. We evaluate the up-sampling results of the selected methods on different anatomical districts and resolutions: obstetric district with 0.5X down-sampling (Fig. 6.9); abdominal district with 0.25X down-sampling (Fig. 6.10).

Table 6.1 summarises a comparison with the PSNR metric, on a test data set of 50 images. *Cubic convolution* has a mean PSNR value of 36.52 and 42.17 for 2X and 4X upsampling, respectively. According to these results, we select *Cubic convolution* as the best method for the up-sampling of US images. This method interpolates the missing lines, without generating artefacts. Also, *Cubic convolution* has a reduced computational cost ($\mathcal{O}(kn)$, with n pixels) that well suits an application in a real-time context.

In comparison, our method improves the results of previous work (Fig. 6.9, Fig. 6.10, Table 6.1), with a mean PSNR value of 37.00 and 44.35 for 2X and 4X super-resolution, respectively. Finally, we underline that 4X super-resolution on the abdominal district has better results than 2X super-resolution on the obstetric district, due to the complexity and variety of each anatomic district data set.

Chapter 7

Kernel-based sampling of US and arbitrary signals

In the context of signal super-resolution, the sampling and approximation of an input signal are relevant to interpolate the input data and reconstruct it on a denser grid. Our starting point is the kernel-based sampling [ZH16], which approximates an input signal on a regular grid (e.g., a 2D or 3D image) as the sum of Gaussian kernels, whose centres are computed through the minimisation of an energy functional. The kernels have a fixed size and the same energy, to provide the same weight to the energy functional. This choice generates artefacts in the sampling of images with complex patterns and limits the approximation accuracy and the distribution of the samples in case of irregularly distributed data.

Starting from the kernel-based sampling, we introduce a set of additional variables, which allow us to improve the quality of the kernel-based sampling in terms of the distribution of the samples, and to further reduce the approximation error. More precisely, we introduce the (σ, α) *kernel-based sampling* (Sect. 7.1), where σ controls the kernel width and α is the vector of coefficients that express the input signal as a linear combination of the Gaussian kernels.

As main contributions with respect to previous work on samplings, we improve the sampling quality and the approximation accuracy, also achieving more accurate feature preservation, with a slightly higher computational cost related to $2n$ additional variables, where n is the number of samples. Analogously to Gaussian Mixture Models (GMMs), we approximate the input signal as a linear combination of radial basis functions (RBFs), whose parameters are optimised through the minimisation of an energy functional; the kernel-based sample corresponds to the mean of the Gaussian function in 1D, the kernel width is equivalent to the standard deviation, and the kernel weight corresponds to the kernel scale. Our method is independent of the dimensionality and spatial organisation of the input data and can be extended to 3D images, unstructured data, solutions to PDEs on arbitrary 2D/3D domains, and vector fields. We apply our method to

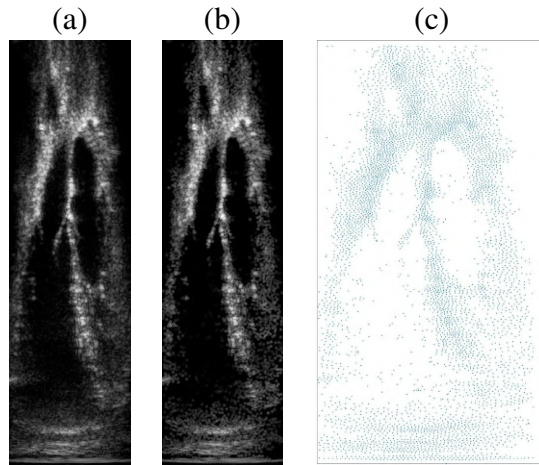


Figure 7.1: (a) Input image of cardiac district (168×580), (b) image reconstruction, and (c) kernel-based sampling with 5K samples.

sample and reconstruct ultrasound 2D images of different anatomical districts. Then, we extend the experimental tests to 2D/3D images (Sect. 7.2).

Related publications

Cammarasana, Simone, and Giuseppe Patanè. "Kernel-Based Sampling of Arbitrary Signals.", *Computer-Aided Design* (2021): 103103. <https://doi.org/10.1016/j.cad.2021.103103> [Journal Paper]

Cammarasana, Simone, and Giuseppe Patanè, "Kernel-Based Sampling of Arbitrary Data.", (2020) - *Eurographics Italian Chapter Conference*. 10.2312/stag.20201252 [Conference Proceeding]

7.1 Kernel-based sampling method and application to US 2D images

The kernel-based sampling [ZH16] approximates an input signal as a linear combination of Gaussian kernel functions, whose centres (i.e., the samples) are computed through the minimisation of the energy functional. We propose a novel method, the (σ, α) *kernel-based sampling*, that

approximates an input signal $C(\mathbf{x})$ on \mathbb{R}^d as a linear combination

$$\begin{cases} C_{\text{recon}}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\sigma}) = k \sum_{j=1}^n \alpha_j G(\mathbf{x}, \boldsymbol{\mu}_j, \sigma_j), \\ G(\mathbf{x}, \boldsymbol{\mu}_j, \sigma_j) := \frac{1}{(\sqrt{2\pi}\sigma_j)^d} \exp\left(-\frac{\|\mathbf{x}-\boldsymbol{\mu}_j\|_2^2}{2\sigma_j^2}\right). \end{cases} \quad (7.1)$$

of functions $G(\mathbf{x}, \boldsymbol{\mu}_j, \sigma_j)$, with centres $\boldsymbol{\mu} = \{\boldsymbol{\mu}_j\}_{j=1}^n$, weights $\boldsymbol{\alpha} := (\alpha_j)_{j=1}^n$, widths $\boldsymbol{\sigma} := (\sigma_j)_{j=1}^n$, and $k = \int_{\Omega} C(\mathbf{x}) ds/n$ as a constant term. Then, the variables $(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\alpha})$ are computed by minimising the energy functional

$$E(\boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\sigma}) = \int_{\Omega} |C(\mathbf{x}) - C_{\text{recon}}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\sigma})|^2 ds, \quad (7.2)$$

In the discrete case, the input signal is known at a set of points $\mathcal{P} = \{\mathbf{x}_i\}_{i=1}^m$ and the integral in Eq. (7.2) is discretized as a finite sum over the input points. Introducing the integrand term $\bar{S}^2(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\alpha})$ of the energy functional (7.2), the minima of $E(\cdot)$ are computed as the roots of its partial derivatives with respect to $\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\alpha}$, i.e.,

$$\begin{aligned} \frac{\partial E(\boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\sigma})}{\partial \boldsymbol{\mu}_i} &= \frac{-2k}{(\sqrt{2\pi})^d \sigma_i^{d+2}} \alpha_i \int_{\Omega} \bar{S}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\alpha}) \exp\left(-\frac{\|\mathbf{x}-\boldsymbol{\mu}_i\|_2^2}{2\sigma_i^2}\right) (\mathbf{x}-\boldsymbol{\mu}_i) ds; \\ \frac{\partial E(\boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\sigma})}{\partial \alpha_i} &= \frac{-2k}{(\sqrt{2\pi}\sigma_i)^d} \int_{\Omega} \bar{S}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\alpha}) \exp\left(-\frac{\|\mathbf{x}-\boldsymbol{\mu}_i\|_2^2}{2\sigma_i^2}\right) ds; \\ \frac{\partial E(\boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\sigma})}{\partial \sigma_i} &= \frac{-2k}{(\sqrt{2\pi}\sigma_i)^d} \int_{\Omega} \bar{S}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\alpha}) \exp\left(-\frac{\|\mathbf{x}-\boldsymbol{\mu}_i\|_2^2}{2\sigma_i^2}\right) \left[\frac{-d}{\sigma_i} + \frac{\|\mathbf{x}-\boldsymbol{\mu}_i\|_2^2}{\sigma_i^3}\right] ds. \end{aligned}$$

We now focus on the properties of the $(\boldsymbol{\sigma}, \boldsymbol{\alpha})$ kernel-based sampling, the evaluation of the energy functional, and its computational cost. The $(\boldsymbol{\sigma}, \boldsymbol{\alpha})$ kernel-based sampling involves $n(d+2)$ variables, i.e., nd variables for the d coordinates of the n samples $\boldsymbol{\mu}$, n variables for the coefficients $\boldsymbol{\alpha}$, and n variables for the Gaussian width $\boldsymbol{\sigma}$. We specialise this method to four variants, by freezing a subset of the free variables in the approximating function (Eq. 7.1):

- $(\boldsymbol{\mu})$ kernel-based sampling (or $(\boldsymbol{\mu})$ -method): we select $\alpha_j = 1, \sigma_j = \sigma, \forall j$, and σ is a constant, thus reducing to the original kernel-based method [ZH16], with nd variables;
- $(\boldsymbol{\sigma})$ kernel-based sampling (or $(\boldsymbol{\sigma})$ -method): we select $\alpha_j = 1, \forall j$ and the corresponding energy functional involves $n(d+1)$ variables;
- $(\boldsymbol{\alpha})$ kernel-based sampling (or $(\boldsymbol{\alpha})$ -method): we select $\sigma_j = \sigma, \forall j$, and σ is constant. The corresponding energy functional involves $n(d+1)$ variables;
- (σ) kernel-based sampling (or (σ) -method): we select $\alpha_j = 1, \sigma_j = \sigma, \forall j$, and σ is a variable. The corresponding energy functional involves $nd+1$ variables.

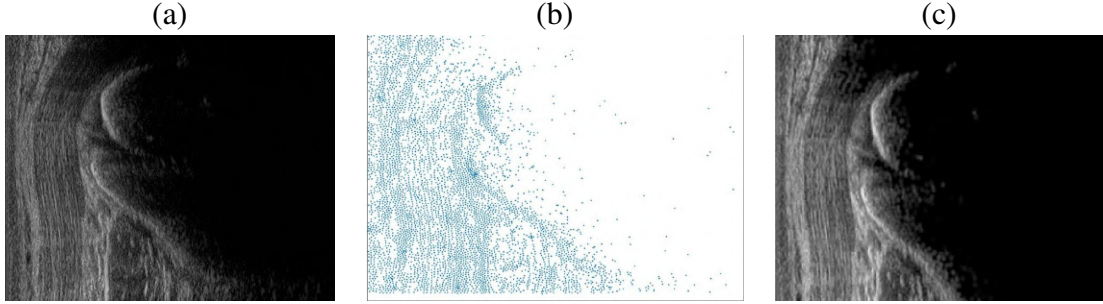


Figure 7.2: (a) Input image of muscle-skeletal district (500×600), (b) image reconstruction, and (c) kernel-based sampling with 5K samples (c).

Signal reconstruction and error metrics Once the samples have been computed, we reconstruct the input signal at any point \mathbf{y} as a linear combination of the kernel functions, by computing $C_{\text{recon}}(\mathbf{y}) := C_{\text{recon}}(\mathbf{y}, \boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\sigma})$ in Eq. (7.1). Then, we evaluate the reconstruction accuracy as the difference $|C_{\text{recon}}(\mathbf{x}_i) - C(\mathbf{x}_i)|$ between the input and the reconstructed signals at \mathbf{x}_i , $\forall i$. Given m input points, we evaluate the *normalised cross correlation* (NCC)

$$\text{NCC} = \frac{\sum_{i=1}^m [C_{\text{recon}}(\mathbf{x}_i) - \bar{C}_{\text{recon}}][C(\mathbf{x}_i) - \bar{C}]}{[\sum_{i=1}^m [C_{\text{recon}}(\mathbf{x}_i) - \bar{C}_{\text{recon}}]^2]^{1/2} [\sum_{i=1}^m [C(\mathbf{x}_i) - \bar{C}]^2]^{1/2}}, \quad (7.3)$$

where \bar{C}_{recon} and \bar{C} are the average values of the reconstructed and input signal respectively, and the *normalised root mean square error* (NRMSE)

$$\text{NRMSE} = \left[\frac{\sum_{i=1}^m [C_{\text{recon}}(\mathbf{x}_i) - C(\mathbf{x}_i)]^2}{\sum_{i=1}^m [C(\mathbf{x}_i)]^2} \right]^{1/2}.$$

The P_k -percentile is defined as the percentage of input points whose reconstruction error is lower than k , i.e.,

$$P_k = \frac{\#\{i : |C_{\text{recon}}(\mathbf{x}_i) - C(\mathbf{x}_i)| < k\}}{m}.$$

In the paper examples, we also visualise the error as the difference between the input and the approximated images, where white corresponds to a null error and black represents the maximum error equal to one (c.f., Fig. 7.4).

Numerical solver and computational cost The minimum of the discrete energy functional is computed through the iterative optimisation method L-BFGS (*Limited-memory Broyden, Fletcher, Goldfarb, Shanno*) [ZBLN97], which finds the roots of the derivative of the energy functional. We briefly recall that L-BFGS is an optimisation algorithm in the family of quasi-Newton methods that approximates the BFGS using a limited amount of computer memory. At each iteration, a small history of the past updates of the position $(\boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\sigma})$ and the gradient of

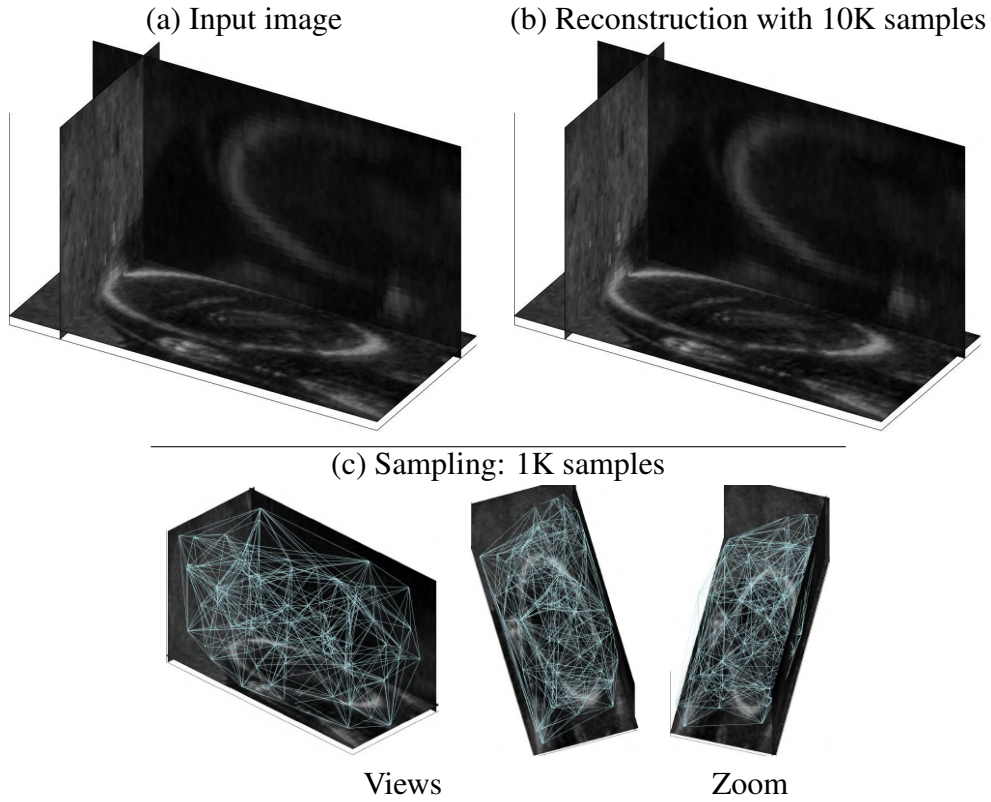


Figure 7.3: (a) input MR image ($156 \times 73 \times 36$), (b) reconstruction (50K samples), and (c) different views and zoom on the sampling with 1K samples.

the energy functional $E(\cdot)$ in Eq. (7.2) is used to identify the direction of steepest descent and to implicitly perform operations requiring vector products with the inverse Hessian matrix. The computational cost of the kernel-based sampling is $\mathcal{O}(u+m)$, where u is the number of variables and m is the number of input points, and the memory storage is $\mathcal{O}(u^2)$.

Experimental tests on US 2D/3D images We apply our kernel-based method to the sampling and reconstruction of US 2D images of different anatomical districts (Fig. 7.1, 7.2), and US 3D image of the obstetric district (Fig. 7.3). Our method permits us to accurately reconstruct the input signal, both in the 2D and 3D case. Even with a reduced number of samples (e.g., Fig. 7.1, 5K samples vs. 100K input points resolution), our method preserves the anatomical structures and geometries of the input image)



















| Input image | Sampling | Reconstruction | Error | Zoom-in | |
|---|--|--|--|--|-----|
|   Colour-map |  Ferrari |  Ferrari |  |  | (a) |
| |  Ferrari |  Ferrari |  |  | (b) |
| |  Ferrari |  Ferrari |  |  | (c) |
| |  Ferrari |  Ferrari |  |  | (d) |

Figure 7.4: Variants comparison: (a), (μ) -method; (b), (σ) -method; (c), (α) -method; (d), (σ, α) -method, with 5K samples. The first column shows the input image (256×256), and the colour-map.

7.2 Further experimental tests

We introduce the experimental results of the kernel-based sampling on 2D/3D images and 3D point clouds.

Kernel-based sampling of 2D images We compare the results of the proposed variants of the kernel-based sampling (Fig. 7.4), and we refer the reader to [CP21] for a complete discussion of the results. The main geometric features of the input image are preserved by all the variants; in particular, the tail and the mane of the horse, or the “Ferrari” label. The reconstruction of the input image with the original kernel-based sampling presents some defects, such as a blurring on the letter “R”, noise around the letter “I”, and a dot-like effect on the grey background of the logo. The (σ) - and (α) -methods provide some improvements to the previous results. In the (σ) -method, the dot-like effect on the grey background is barely visible. The characters are

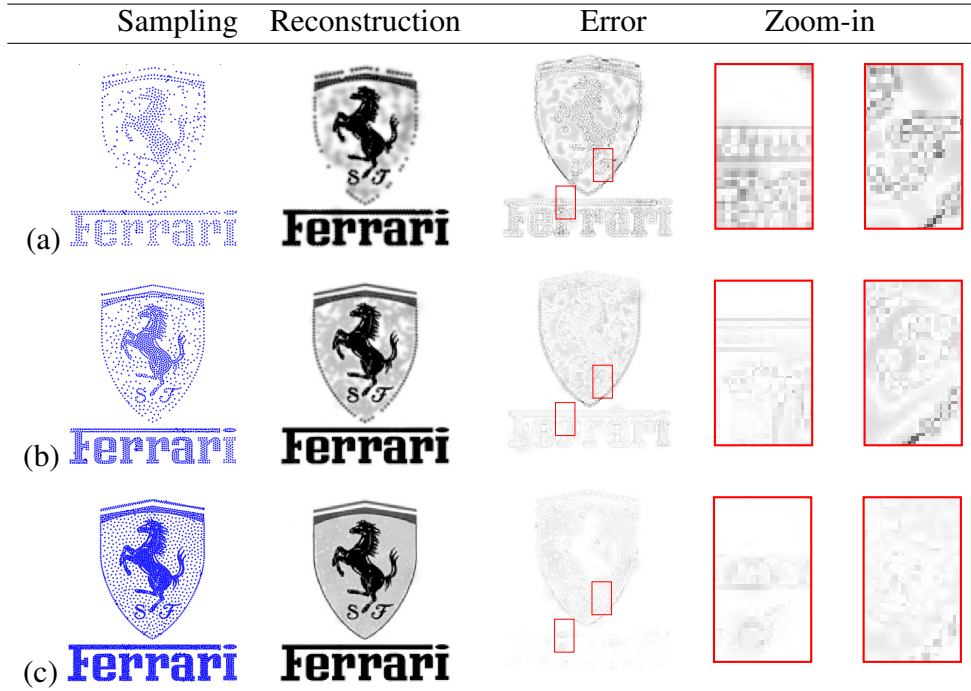


Figure 7.5: Accuracy of the (σ, α) -method, with a different number of samples: (a) $n = 1K$; (b) $n = 2K$; (c) $n = 4K$.

well reconstructed, apart from the letter “R”, which is still blurred. The noise around the letter “I” is not present.

A different reconstruction accuracy of the “R” letters (i.e., the first one is more blurred than the second and the third ones) depends on the overall number of samples and on the number of samples that belong to this area of the image; in (a), 190 samples are placed in the first “R” letter, 273 and 282 samples are placed in the second and third “R” letter respectively, thus leading to a different approximation and reconstruction of this area. The number of samples for each “R” letter depends on two factors: the samples’ initialisation and the iterative optimisation of the energy functional. Since the optimisation of the energy functional is deterministic and the samples’ initialisation is stochastic [ZH16], a different initialisation of the method leads to a larger or smaller number of samples for each “R” letter, and consequently, to a different reconstruction of the corresponding area of the image.

The (α) -method is affected by a dot-like effect on the grey background; furthermore, the “Ferrari” label looks more jagged and irregular. The reconstruction accuracy of the (σ, α) -method is very good; in fact, there is no blurring or dot-like effects, the background is very uniform, and the grey distribution of the original image is preserved. Furthermore, all the letters of the “Ferrari” label are perfectly reconstructed. Considering the error distribution between the input and the reconstructed images, the (μ) -method has a visible error on most of the characters of

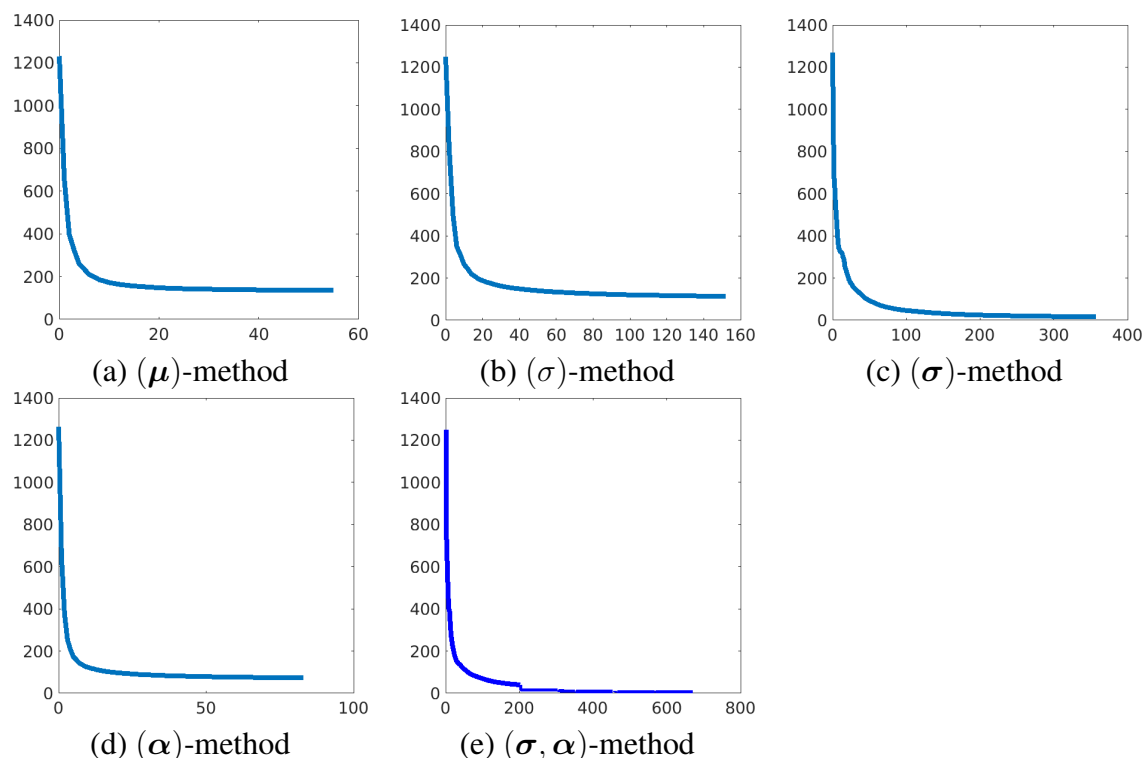


Figure 7.6: Concerning Fig. 7.4, we report the convergence of the iterative minimisation, in terms of the objective function value (y -axis) with respect to the number of iterations (x -axis).

“Ferrari”; in particular, on the letter “R”. Also, the error is higher on the external boundary of the shapes. Then, the error of the (σ) - and (α) -methods is significantly reduced (e.g., in the logo background and the letter “R”), and it is barely visible in the (σ, α) -method, including the “Ferrari” label.

We conclude that the (σ) -method provides better results than (α) -method, particularly in the background reconstruction; indeed, the use of the σ variables improves the quality of the approximation of the input image with respect to the α variables, despite the number of variables remains unchanged. The (σ, α) -method gives the best results, with an excellent sampling, approximation, and reconstruction of the input image.

Approximation accuracy In Table 7.1, we present a comparison of the five kernel-based sampling methods, in terms of the error metrics described in Sect. 7.1. First of all, each variant of the kernel-based sampling further reduces the numerical value of the objective function with respect to the (μ) -method. The minimum of the energy functional is computed on a larger trust region. The (μ) -method has an objective function value of 150, a $P_{0.10}$ value of 94.2% and a NCC value of 0.988. The (σ) -method has an objective function value of 112, it improves the $P_{0.10}$ value to

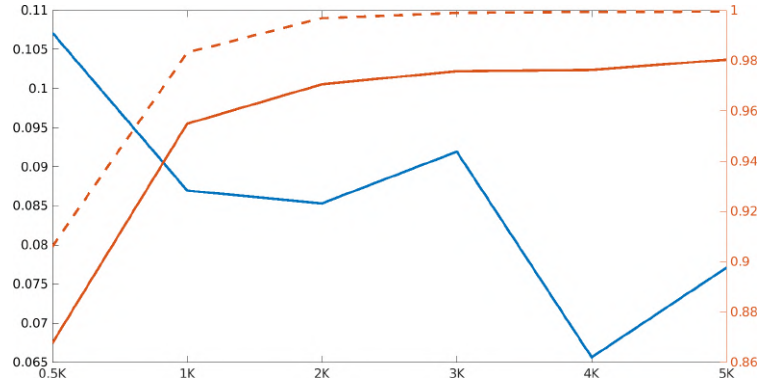


Figure 7.7: NCC error (y -axis, right): (μ) - (red continuous line) and (σ, α) - (red dashed line) kernel-based samplings; Hausdorff distance (y -axis, left) of the corresponding samples (blue line), with respect to their number (x -axis). Image: Dante, 256×256 .

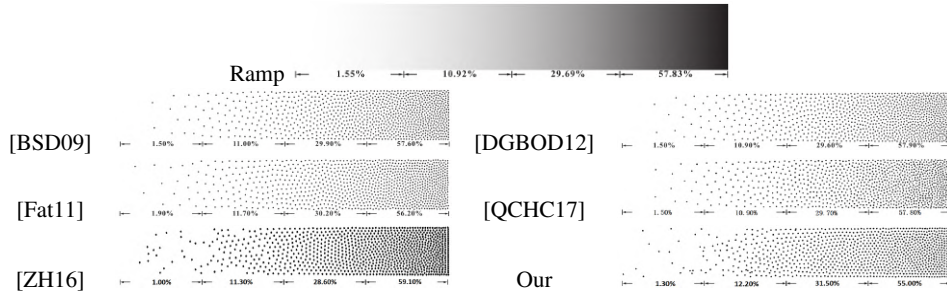


Figure 7.8: Ramp image (192×75) and sampling (1K samples) comparison among state-of-the-art methods; the percentages of each quarter show the number of samples in that portion of the grid, compared with the ink density of the input image.

95.3%, and the NCC value to 0.990. The (σ) -method has an objective function value of 29.8; it has very good $P_{0.05}$ and $P_{0.10}$ values, 96.4% and 98.8% respectively. The (α) -method has an objective function value of 73; the NCC, the $P_{0.05}$, and the $P_{0.10}$ values are similar to the ones of the (σ) -method. Finally, the (σ, α) -method significantly improves all the metrics for the error evaluation, since it has an objective function value of 6.65, the $P_{0.10}$ value is 99.5%, and the NCC value is 0.998.

The (σ) -method has better results than the original method, with the addition of one variable only. The (σ) - and (α) -methods have worse results than the (σ, α) -method, and the (σ) -method has better results than the (α) -method, despite the same number of variables. Finally, the (σ, α) -method significantly improves the (μ) -method under all the metrics.

Approximation accuracy with respect to the number of samples Since the (σ, α) -method has the highest approximation quality, we further analyse its accuracy with respect to the number



Figure 7.9: Comparison of the reconstructed ramp image, between (a) [ZH16], and (b) ours.






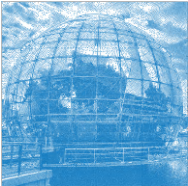


| Image | Sampling | Reconstruction | Error |
|--|--|---|--|
|  |  |  |  |
| (a) | | | |
|  |  |  |  |
| (b) | | | |

Figure 7.10: Sampling initialisation on (a) Supermario (256×256 input points, 16K samples, $m = 4$), (b) Biosphere (512×512 input points, 50K samples, $m = 16$).

of samples. Selecting a different number of samples (i.e., from 1K to 4K), the blurring and the dot-like effects are more accentuated with a lower number of samples; however, the overall features (e.g., the horse shape, the “Ferrari” label), are well reconstructed. The reconstruction error decreases as the number of samples increases; especially on the letter “R” and on the background of the logo (Fig. 7.5).

Comparing the error metrics (Table 7.2), the (σ, α) -method is very accurate, even when we use only 1K samples (i.e., the 1.5% of the input pixels). In fact, 77% of the points have a reconstruction error lower than 0.05, and 85% of the points have an error lower than 0.10. We underline that the accuracy of the (σ, α) -method with 2K samples is comparable to the (μ) -method with 5K samples (Table 7.1).

Computation time Table 7.3 shows the variation of the execution time and the number of iterations of the L-BFGS algorithm to converge, compared to the original kernel-based sampling (i.e., the (μ) -method). Tests have been performed on a workstation with 3.1 GHz Dual-Core Intel Core i7, and 16GB RAM. The (μ) -method takes 67 iterations and 24 seconds to converge to the

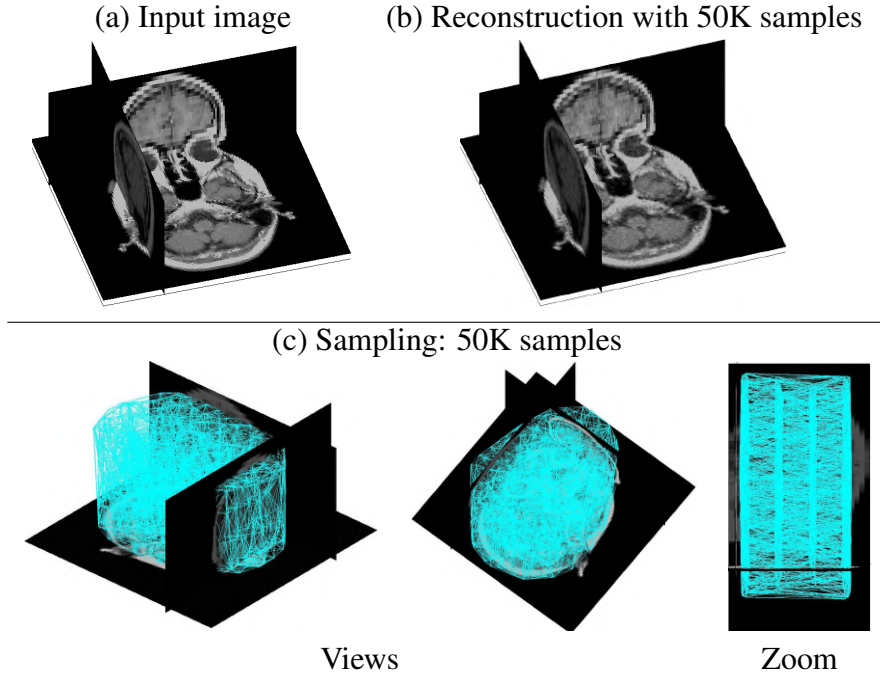


Figure 7.11: (a) input MR image ($128 \times 128 \times 24$), (b) reconstruction, and (c) different views and zoom on the sampling with 50K samples.

solution. The (σ) -method increases the execution time by 2.3 times and the number of iterations to 149. The (σ) -method has a very high execution time, due to the much larger number of variables, and the slow convergence of the algorithm (357 iterations). The (α) -method takes only 83 iterations, thus resulting very close to the original kernel-based sampling, but its execution time is increased by 1.9 times. The (σ, α) -method increases the execution time by 9.1 times, and it takes 674 iterations to converge to the solution.

Fig. 7.6 shows the value of the objective function (y -axis) with respect to the number of iterations (x -axis) for the five methods. The (σ) - and the (σ, α) -methods have a slower convergence to zero with respect to the other variants of the kernel-based sampling. According to Table 7.4 and selecting $n = 5K$ samples, the allocated memory increases of about four times, when passing from the (μ) - to the (σ, α) -method. Table 7.5 shows the execution time and the number of iterations of the (σ, α) -method, when varying the number of samples. Decreasing the number of samples the number of iterations increases, as the solver converges more slowly.

Comparison between samplings We compare the (μ) and (σ, α) kernel-based samplings in terms of their Hausdorff distance and the approximation accuracy of the reconstructed signals. We recall that the *symmetric Hausdorff distance* of two point sets \mathcal{X}, \mathcal{Y} is defined as $d(\mathcal{X}, \mathcal{Y}) := \max\{d_{\mathcal{X}}(\mathcal{Y}), d_{\mathcal{Y}}(\mathcal{X})\}$, with $d_{\mathcal{X}}(\mathcal{Y}) := \max_{\mathbf{x} \in \mathcal{X}} \{\min_{\mathbf{y} \in \mathcal{Y}} \{\|\mathbf{x} - \mathbf{y}\|_2\}\}$ *one-side*

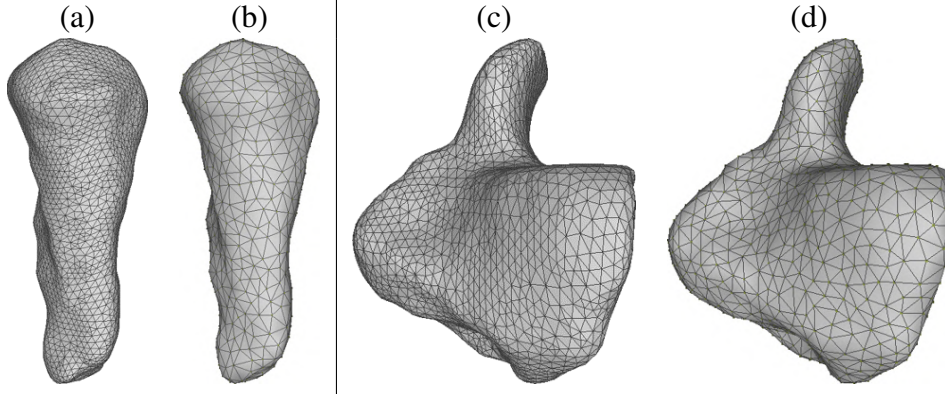


Figure 7.12: (a) input point cloud (30k points) and (b) our sampling (5k samples) of the ulna; (c) input point cloud (15k points) and (d) our sampling (3k samples) of the scaphoid .

Table 7.1: With reference to Fig. 7.4, we report the approximation accuracy of the five kernel-based sampling variants, with the best results in bold.

| Method | (μ) | (σ) | (σ) | (α) | (σ, α) |
|---------------------------|---------|------------|------------|------------|--------------------|
| Objective function | 149.9 | 112.0 | 29.8 | 73.0 | 6.65 |
| NCC | 0.988 | 0.990 | 0.996 | 0.992 | 0.998 |
| NRMSE | 0.048 | 0.044 | 0.025 | 0.043 | 0.019 |
| $P_{0.05}$ | 86.5% | 86.5% | 96.4% | 86.3% | 96.9% |
| $P_{0.10}$ | 94.2% | 95.3% | 98.8% | 95.2% | 99.5% |

Hausdorff distance.

In Fig. 7.7, we show the NCC error of the (μ) - and the (σ, α) -methods, and the Hausdorff distance (y -axis) between the sampling of the two methods, when varying the number of samples from 500 to 5K (x -axis). The variation of the values of the Hausdorff distance shows that the samplings of the two methods remain different as we increase the number of samples. The NCC trend shows that the (σ, α) kernel-based sampling remains more accurate than the standard sampling. In Table 7.6, we report the Hausdorff distance between the sampling of the four variants (b-e) and of the (μ) -method (i.e., the kernel-based sampling [ZH16]); the sampling of the (σ) -method is the closest one to the original kernel-based sampling, while the sampling of the (σ) -method is the farthest.

We also compare our approach with state-of-the-art sampling methods on an intensity-increasing image, counting the number of samples for each quarter of the ramp image (Fig. 7.8). Our method achieves results analogous to [ZH16] and is comparable with state-of-the-art methods, in terms of sampling. Furthermore, it better reconstructs the input signal with respect to [ZH16] (Fig. 7.9), improving the preservation of grey-scale values and reducing the scattering effect. Considering the quantitative metrics of the reconstructed images, the NCC value is 0.987 and

Table 7.2: With reference to Fig. 7.5, we report the error metrics for the (σ, α) -method with a different number of samples; best results are in bold.

| Samples | 5K | 4K | 3K | 2K | 1.5K | 1K |
|---------------------------|--------------|-------|-------|-------|-------|-------|
| Objective function | 6.65 | 17.5 | 50.1 | 123.2 | 200.2 | 350.4 |
| NCC | 0.998 | 0.997 | 0.995 | 0.989 | 0.982 | 0.971 |
| NRMSE | 0.019 | 0.025 | 0.033 | 0.048 | 0.064 | 0.088 |
| $P_{0.05}$ | 96.9% | 92.8% | 89.4% | 83.8% | 80.6% | 77.4% |
| $P_{0.10}$ | 99.5% | 99.4% | 98.3% | 94.6% | 90.3% | 84.7% |

Table 7.3: With reference to Fig. 7.4, we report the increment of the execution time and the number of iterations for the five variants, with respect to the execution time $T = 24s$ (in seconds s) of the (μ) -method.

| Method | (μ) | (σ) | (σ) | (α) | (σ, α) |
|---------------------------|---------|------------|------------|------------|--------------------|
| Execution Time [s] | T | $2.3T$ | $8.4T$ | $1.9T$ | $9.1T$ |
| Iterations | 67 | 149 | 357 | 83 | 674 |

the NRMSE value is 0.084 for our method, while these values are 0.946 and 0.151 respectively in [ZH16].

Kernel-based sampling and least-squares approximation Once the samples have been computed, we reconstruct the input signal as

$$C_{LS}(\mathbf{x}) := \sum_{j=1}^n \gamma_j G(\mathbf{x}, \boldsymbol{\mu}_j, \sigma_j) \stackrel{\text{Eq. (7.1)}}{=} C_{\text{recon}}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\gamma}, \boldsymbol{\sigma}), \quad (7.4)$$

where $\boldsymbol{\gamma}$ is the $n \times 1$ coefficients' vector that solve the least-squares linear system $\mathbf{G}\boldsymbol{\gamma} = \mathbf{C}$. Here, \mathbf{G} is the $m \times n$ Gram matrix associated with the Gaussian kernel, evaluated at the m points $(\mathbf{x}_i)_{i=1}^m$, and at the n samples $(\boldsymbol{\mu}_j)_{j=1}^n$, $\mathbf{C} := (C(\mathbf{x}_i))_{i=1}^m$ is the $m \times 1$ vector associated with the input signal. According to Table 7.7, the least-squares reconstruction has a better accuracy than the standard kernel-based reconstruction through Eq. (7.1) when using the (μ) -method, while the two reconstruction approaches have the same accuracy when the (σ, α) -method is used.

Sampling initialisation For the sampling initialisation, our goal is to increase the number of samples and the resolution of the input image, overcoming memory limits that previous methods might encounter when computing/allocating the partial derivatives and evaluating the minimum value of the energy functional (Table 7.4). To this end, we split the input image into a set of images, and we work independently on each of them. More precisely, given a 2D image Ω ,

Table 7.4: Memory allocation of the kernel-based sampling, when using $n = 5K$ samples; each variable is stored as a double precision number.

| Method | (μ) | (σ) | (σ) | (α) | (σ, α) |
|-------------------------------|---------|------------|------------|------------|--------------------|
| Memory allocation [GB] | 0.74 | 0.74 | 1.68 | 1.68 | 2.98 |

Table 7.5: With reference to Fig. 7.5, we report the reduction of the execution time and the number of iterations for the (σ, α) -method with a different number of samples, with respect to the execution time $T = 216s$ with 5K samples.

| Samples | 5K | 4K | 3K | 2K | 1.5K | 1K |
|---------------------------|-----|-----|--------|--------|--------|--------|
| Execution time [s] | T | T | $0.6T$ | $0.4T$ | $0.4T$ | $0.2T$ |
| Iterations | 657 | 896 | 785 | 929 | 891 | 1052 |

we split it into m 2D images $(\Omega_i)_{i=1}^m$, and the number of samples of each sub-image is defined according to its grey intensity $C_{\Omega_i} = \int_{\Omega_i} C(\mathbf{x}) ds$ as $n_{\Omega_i} = (C_{\Omega_i}/C_{\Omega})n_{\Omega}$, where n_{Ω} is the overall number of samples. In this way, we avoid too many samples being placed in those regions where the white colour is predominant.

To further improve the reconstructed image, we add a small overlay to the sub-images to be sampled: Supermario (Fig. 7.10(a), 256×256) is divided into four sub-images of 132×132 , with an overlap of 132×8 for each couple of adjacent sub-images, and has been sampled with an overall number of $n = 16K$ samples. Similarly, the Biosphere image (512×512) is divided into 16 sub-images of 136×136 resolution, with an overlap of 136×16 for each couple of adjacent sub-images, with $n = 50K$ samples. Then, the reconstructed image in the overlapped area is computed as the average between the two reconstructed sub-images.

For both examples, the sampling and the reconstructed image preserve the main features of the two subjects. In Supermario, the samples are well localised in dark areas (e.g., the hat, the body, the moustaches), and all the features are well reconstructed (e.g., the letter ‘‘M’’). The reconstruction error is very low and uniform. The Biosphere is more difficult to be sampled and reconstructed, due to its high resolution, and the presence of complex geometries and their overlaps, e.g., the sphere, clouds, plants, and shadows on the sea. The approximation error is lower on the sphere structure, which is well reconstructed, and generally higher on the clouds and the sea shadows. According to Table 7.8, the approximation accuracy is very good for both the examples, due to a high number of selected samples. In particular, the NCC value of Supermario is 0.9996, compared to the NCC value of 0.9954 when using 5K samples.

Kernel-based sampling of 3D data Fig. 7.11 and Table 7.9 show the results of the sampling and reconstruction of a volumetric MR image, with size $128 \times 128 \times 24$; the initialisation (Sect. 7.2) with 50K samples is applied to 8 sub-images of size $128 \times 128 \times 3$. The sampling covers the entire brain, and the reconstructed image accurately approximates the main features of the

Table 7.6: With reference to Fig. 7.4, we report the Hausdorff distance normalised with the diagonal of the image, between the four variants (b-e) and the original kernel-based sampling (i.e., the $(\boldsymbol{\mu})$ -method).

| Method | (σ) | $(\boldsymbol{\sigma})$ | $(\boldsymbol{\alpha})$ | $(\boldsymbol{\sigma}, \boldsymbol{\alpha})$ |
|--------------------|------------|-------------------------|-------------------------|--|
| Hausdorff distance | 0.0157 | 0.0104 | 0.0117 | 0.0105 |

Table 7.7: NCC values for the reconstruction associated with the $(\boldsymbol{\mu})$ -, and $(\boldsymbol{\sigma}, \boldsymbol{\alpha})$ - methods and applied to Pinocchio 256×256 .

| Num. samples | Reconstr. method | $(\boldsymbol{\mu})$ | $(\boldsymbol{\sigma}, \boldsymbol{\alpha})$ |
|--------------|------------------------------|----------------------|--|
| 2K | Gaussian kernels (Eq. (7.1)) | 0.6612 | 0.7944 |
| | C_{LS} (Eq. (7.4)) | 0.7962 | 0.7944 |
| 5K | Gaussian kernels (Eq. (7.1)) | 0.8805 | 0.9637 |
| | C_{LS} (Eq. (7.4)) | 0.9372 | 0.9637 |

input data. In this test, the samples are denser where the input signal is higher (e.g., white parts of the image), and our method improves the results of [ZH16], as the NCC value increases from 0.82 to 0.91, and the $P_{0.10}$ metrics increases from 0.1 to 0.4. Due to a larger number (i.e., 250K) of variables, the execution time (2700s) increases with respect to [ZH16] (i.e., 150K variables).

Kernel-based sampling of 3D point clouds For the sampling of an arbitrary point cloud, the samples are computed through the minimisation of the energy functional $E(\boldsymbol{\mu}) = w_1 E_1(\boldsymbol{\mu}) + w_2 E_2(\boldsymbol{\mu}) + E_3(\boldsymbol{\mu})$, where

$$E_1(\boldsymbol{\mu}) = \int_{\Omega} \sum_{i=1}^{n_1} \frac{1}{(\sqrt{2\pi}\sigma)^d} \exp\left(-\frac{\|\mathbf{x} - \boldsymbol{\mu}_i\|_2^2}{2\sigma^2}\right) d\mathbf{s};$$

$$E_2(\boldsymbol{\mu}) = \int_{\Omega} \sum_{i=1}^{n_2} \frac{1}{(\sqrt{2\pi}\sigma)^d} \exp\left(-\frac{\|\mathbf{x} - \boldsymbol{\mu}_i\|_2^2}{2\sigma^2}\right) d\mathbf{s};$$

$$E_3(\boldsymbol{\mu}) = \sum_{s=1}^n \sum_{i=1}^{n_3} \frac{1}{(\sqrt{2\pi}\sigma)^d} \exp\left(-\frac{\|\boldsymbol{\mu}_s - \boldsymbol{\mu}_i\|_2^2}{2\sigma^2}\right).$$

The terms $E_1(\boldsymbol{\mu})$ and $E_2(\boldsymbol{\mu})$ force the samples to be close to the input points. $E_3(\boldsymbol{\mu})$ forces the samples not to overlap. Here, w_1 and w_2 are the weights of the components of the energy functional. The difference between the definition of $E_1(\boldsymbol{\mu})$ and $E_2(\boldsymbol{\mu})$ is in the way the neighbours are computed; in $E_1(\boldsymbol{\mu})$, we search the n_1 nearest samples for each input point, and in $E_2(\boldsymbol{\mu})$ we search the n_2 nearest input points for each sample. The derivative of the terms E_1 , E_2 , and E_3

Table 7.8: With reference to Fig. 7.10, we report the approximation accuracy of the sampling optimisation.

| Image | Supermario | Biosphere |
|--------------|------------|-----------|
| NCC | 0.9996 | 0.985 |
| NRMSE | 0.0125 | 0.0884 |
| $P_{0.05}$ | 99.3% | 78.4% |
| $P_{0.10}$ | 99.9% | 95.3% |

Table 7.9: With respect to Fig. 7.11, we report the quantitative metrics and the execution time of our method and the kernel-based sampling [ZH16].

| Method | [ZH16] | Ours |
|--------------------------|--------|-------------|
| NCC | 0.82 | 0.91 |
| NRMSE | 1.00 | 0.83 |
| $P_{0.05}$ | < 0.1 | 0.18 |
| $P_{0.10}$ | < 0.1 | 0.59 |
| Execution time[s] | 1020 | 2700 |

are

$$\left\{ \begin{array}{l} \frac{\partial E_h(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}_i} = \frac{-1}{(\sqrt{2\pi})^d \sigma^{d+2}} \int_{\Omega} \exp\left(-\frac{\|\mathbf{x} - \boldsymbol{\mu}_i\|_2^2}{2\sigma^2}\right) (\mathbf{x} - \boldsymbol{\mu}_i) d\mathbf{s}, \quad h = 1, 2, \\ \frac{\partial E_3(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}_i} = \frac{1}{(\sqrt{2\pi}\sigma)^d} \sum_{s=1}^n \left[\exp\left(-\frac{\|\boldsymbol{\mu}_s - \boldsymbol{\mu}_i\|_2^2}{2\sigma^2}\right) (-2)(\boldsymbol{\mu}_s - \boldsymbol{\mu}_i) + \sum_{\substack{h=1 \\ h \neq i}}^{n_3} \exp\left(-\frac{\|\boldsymbol{\mu}_s - \boldsymbol{\mu}_h\|_2^2}{2\sigma^2}\right) 2(\boldsymbol{\mu}_i - \boldsymbol{\mu}_h) \right]. \end{array} \right.$$

Our model is applied to the 3D point cloud of an ulna and a scaphoid (Fig. 7.12); our sampling well preserves the main geometrical features of the input point cloud. For further experimental tests on 2D/3D point cloud sampling, we refer the reader to [CP20].

PART IV

Discussion and future work

Chapter 8

Conclusion and future work

Denoising and super-resolution are part of a broader context of healthcare data management, ranging from data acquisition to providing one or more feedback to the physician (Fig. 8.1). In this context, we discuss the results of the proposed denoising and super-resolution and future work.

Conclusion

The denoising of US signals is relevant for post-processing and visual evaluation by medical experts. Several works show the benefits of denoising for classification [CSJ⁺10], feature extraction [IKM08], segmentation [HLZ17], and quantitative analysis [KBSA00], for a more accurate estimation of patient diseases (e.g., the dimension of a tumour, the identification of a certain tissue).

Denoising aims to achieve the best compromise between noise removal, feature preservation, and real-time execution. Fast handcrafted methods [GK19] have lower results in terms of noise removal and edges enhancement; GPU-based methods [FM16] have higher hardware requirements than our method; other denoising methods [XZZ18] have good results in terms of noise removal, but they can not reach a real-time implementation, due to high computational cost. To bypass these limits, we have presented a novel deep learning framework with HPC tools for the real-time denoising of US images. The use of deep learning in our framework allows us to apply state-of-the-art denoising methods to the real-time context of US image processing, overcoming the computational limits and preserving the effectiveness of these methods in terms of noise removal and anatomical feature preservation. Our framework is general enough to be applied to different anatomical districts and noise levels, and supports the tuning of the denoising parameters (e.g., tuned-WNNM) to obtain the best denoising results, as this tuning only affects the

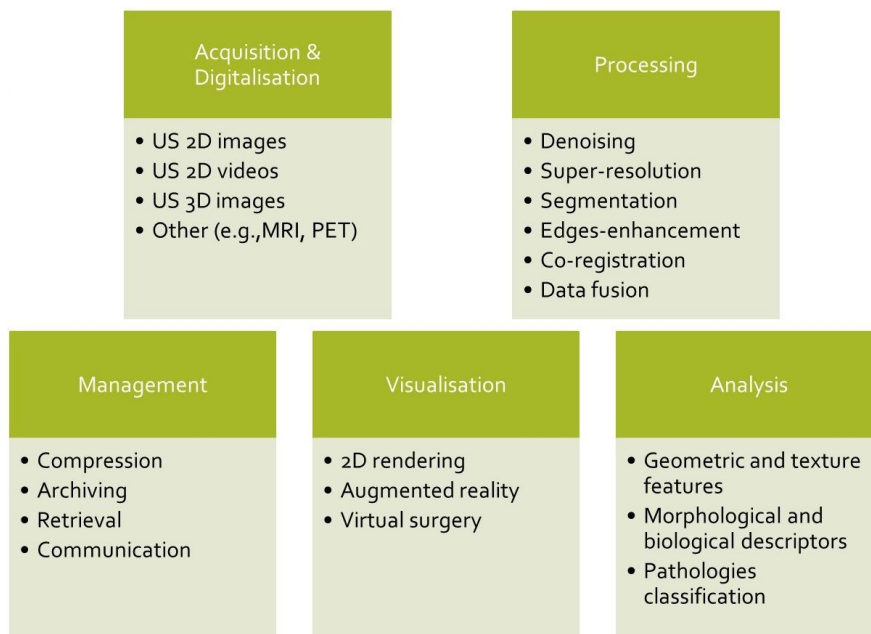


Figure 8.1: Healthcare: biomedical data management chain.

training phase, while the real-time computation of the denoised image is performed through the prediction of the network. The development of our framework to US 2D videos and 3D images shows the generality of our approach, even allowing the extension of inherently non-real-time methods (e.g., Vidosat) in the context of real-time US processing. Finally, our low-rank method shows that AI techniques can also be applied within classical denoising methodologies, extending learning-based methods to the processing of low-dimensional data extracted from images with traditional decomposition and representation techniques. The quantitative and qualitative validations on US and synthetic signals confirm the effectiveness of the proposed framework.

Super-resolution of US images is relevant in many medical specialities, such as oncology [GAD12], neurology [SEDP19], nephrology [FZI⁺17], and rheumatology [HCJB⁺18]. We have introduced a novel deep learning framework for the real-time super-resolution of US images, which improves the quality of the up-sampling of a selected state-of-the-art algorithm by training a neural network to match the target high-resolution image. Our framework extends state-of-the-art up-sampling algorithms to the US application, also improving the accuracy of the super-resolution and reducing artefacts and blurring in real-time execution. In the context of signal approximation, we have presented our novel method of kernel-based sampling that optimises the centres, supports, and coefficients of the reconstructed signal, with an application to US 2D and 3D images. Our method improves the signal approximation and reconstruction, by preserving the original characteristics of the input signal in terms of features and grey intensities.

Finally, through learning and high-performance computing, the proposed denoising and super-

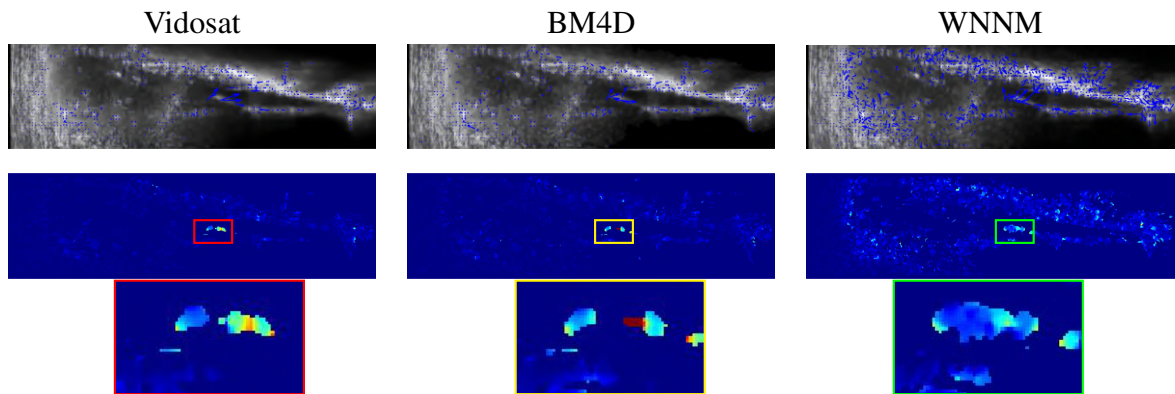


Figure 8.2: Displacement analysis of the mitral valve in the cardiac district: optical flow (1st row), optical magnitude (2nd row), magnitude magnification (3rd row). The denoising through specific algorithms allows us different characterizations of the morphological features of the anatomical part.

resolution algorithms have been specialised to different anatomical districts, and to shift the computational demand to centralised hardware resources with a real-time execution of the network's prediction on local devices.

Limitations As main limitations, our denoise and super-resolution models use learning methods trained on large data sets of US images and videos acquired from various anatomical districts and made available by Esaote. The used data sets are private while the availability of public data may be subject to corporate or hospital privacy constraints and it is often limited in terms of quantity and variety of images. Qualitative validation needs one or more industry/clinical experts and possibly multi-disciplinary skills, such as physicians specialised in different anatomical districts (e.g., cardiologist, orthopaedic). Furthermore, qualitative reviews are difficult to compare and reproduce within the scientific community. While using HPC hardware and methodologies for training can be done offline, the testing part would require a cloud infrastructure for managing the communication between local machines and central hardware; this would require a reorganization of procedures, hardware, and information flow. Finally, the transition from synthetic (training) to biomedical (tests) data sets for the application of learning models can lead to losses in accuracy due to the different characteristics of the data in terms of complexity, geometry, texture properties of the image.

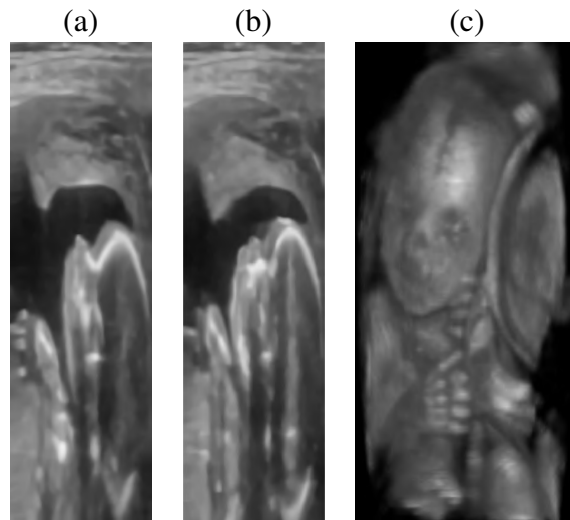


Figure 8.3: (a,b) Two different slices of denoising and super-resolution of Fig. 4.1 with our methods at a resolution of $624 \times 292 \times 36$; (c) Ray-tracing of the volume.

Future work

Denoising and super-resolution are also applied as pre-processing for improving visualisation, segmentation, and predictive analysis. The analysis of the optical flow [HS81] of US videos by evaluating the movement of relevant anatomical features (Fig. 8.2) shows us that the denoising through Vidosat and BM4D improves the characterisation of the movement of the mitral valve, with a better estimation of its movement during the opening/closing phases. Combining denoising with super-resolution (Fig. 8.3) improves (i) the quality of the rendering of US images through ray-casting [Rot82] (Sect. 4.2) and (ii) the accuracy of the segmented contours of breast tumours; then, the extraction of geometric and texture-based parameters through our preliminary method from both the US 3D image and the mesh allows us to classify tumours according to benign against malignant labels (Fig. 8.4).

Starting from these preliminary results, as future work we plan to develop novel methods for the analysis and processing of US signals, where denoising and super-resolution are applied as pre-processing operations. In particular, the main further research activities are:

- *Segmentation* of the region of interest, for the detection of the boundaries of a tumour [ICM21], or the shape of the heart wall [LCW20];
- *Morphological and quantitative* estimation of anatomical parameters for the quantification and monitoring of the temporal variation of anatomical parameters (e.g., bones erosion progression, cartilage damage) with the integration of volume-based approaches and

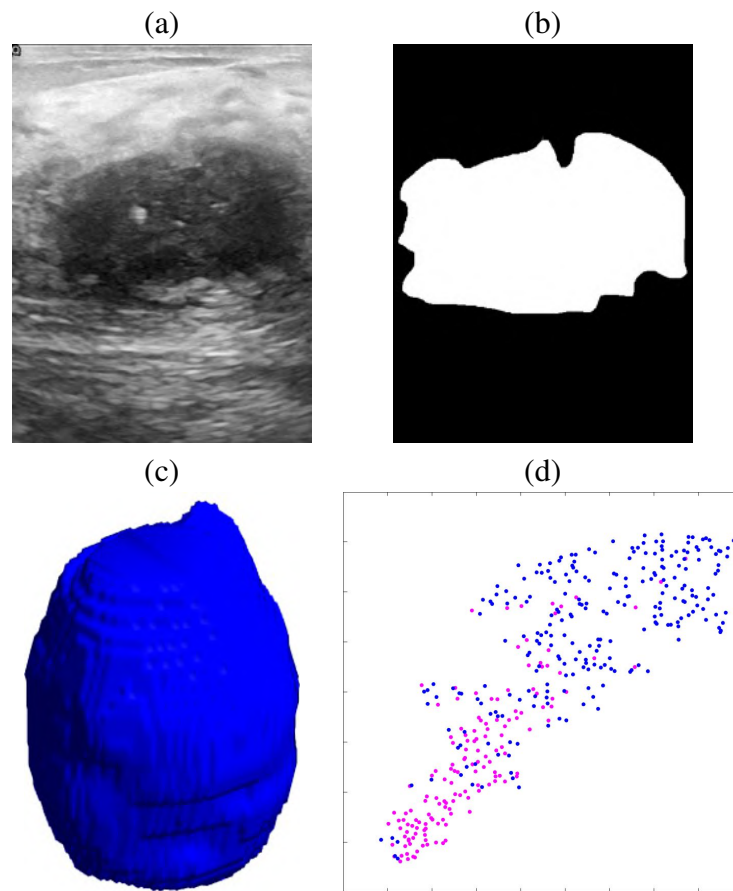


Figure 8.4: (a) US slice (400×300) of a volumetric image ($400 \times 300 \times 36$) of breast district; (b) segmentation mask; (c) extracted mesh, 15K points, 30K triangles; (d) classification of breast tumours in benign (blue) and malignant (magenta).

surface-based approaches (i.e., defined on the segmented surfaces extracted from US volume). Main examples include the pleural line properties (e.g., thickness, roughness) in lungs [WZH⁺21], or geometric features of segmented tumours in the breast anatomical district [TMDCSN21];

- *Classification and prediction* of the image according to pre-defined classes, such as classification of benign vs. malignant tumour of the thyroid gland [WJJ⁺16], or prediction of coronary artery disease [RFG⁺18];
- *Clinical validation* of the denoising and super-resolution results performed by Esaote quality department and expert radiologists, based on an interdisciplinary approach that involves engineering, medical science, physics, and computer science.

Bibliography

- [AA92] Mostafa Analoui and Jan Allebach. New results on reconstruction of continuous-tone from halftone. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages 313–316, 1992.
- [AC17] Byeongyong Ahn and Nam Ik Cho. Block-matching convolutional neural network for image denoising. *arXiv:1704.00524*, 2017.
- [AEB06] Michal Aharon, Michael Elad, and Alfred Bruckstein. K-svd: an algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, 2006.
- [AFAL06] Santiago Aja-Fernández and Carlos Alberola-López. On the estimation of the coefficient of variation for anisotropic diffusion speckle filtering. *IEEE Transactions on Image Processing*, 15(9):2694–2701, 2006.
- [AHRH⁺17] Hadeal Abdulaziz Al Hamid, Sk Md Mizanur Rahman, M Shamim Hossain, Ahmad Almogren, and Atif Alamri. A security model for preserving the privacy of medical big data in a healthcare cloud using a fog computing facility with pairing-based cryptography. *IEEE Access*, 5:22313–22328, 2017.
- [AMP⁺11] Martino Alessandrini, Simona Maggio, Jonathan Porée, Luca De Marchi, Nicolo Speciale, Emilie Franceschini, Olivier Bernard, and Olivier Basset. A restoration framework for ultrasonic tissue characterization. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 58(11):2344–2360, 2011.
- [ANMM⁺17] Mohamed Abdel-Nasser, Jaime Melendez, Antonio Moreno, Osama A Omer, and Domenec Puig. Breast tumor classification in ultrasound images using texture analysis and super-resolution methods. *Engineering Applications of Artificial Intelligence*, 59:84–92, 2017.

- [ANO16] Mohamed Abdel-Nasser and Osama Ahmed Omer. Ultrasound image enhancement using a deep learning architecture. In *International Conference on Advanced Intelligent Systems and Informatics*, pages 639–649. Springer, 2016.
- [ASR⁺15] Emran Mohammad Abu Anas, Alexander Seitel, Abtin Rasouljan, Paul St John, David Pichora, Kathryn Darras, David Wilson, Victoria A Lessoway, Ilker Hacihaliloglu, Parvin Mousavi, Robert Rohling, and Purang Abolmaesumi. Bone enhancement in ultrasound using local spectrum variations for guiding percutaneous scaphoid fracture fixation procedures. *International Journal of Computer Assisted Radiology and Surgery*, 10(6):959–969, 2015.
- [ATK⁺21] Sofie Bech Andersen, Iman Taghavi, Hans Martin Kjer, Stinne Byrholdt Sjøgaard, Carsten Gundlach, Vedrana Andersen Dahl, Michael Bachmann Nielsen, Anders Bjorholm Dahl, Jørgen Arendt Jensen, and Charlotte Mehlin Sørensen. Evaluation of 2d super-resolution ultrasound imaging of the rat renal vasculature using ex vivo micro-computed tomography. *Scientific reports*, 11(1):1–13, 2021.
- [AZA18] Jamil Azzeh, Bilal Zahran, and Ziad Alqadi. Salt and pepper noise: effects and removal. *International Journal on Informatics Visualization*, 2(4):252–256, 2018.
- [BAPMCG⁺19] Xavier P Burgos-Artizzu, Álvaro Perez-Moreno, David Coronado-Gutierrez, Eduard Gratacos, and Montse Palacio. Evaluation of an improved tool for non-invasive prediction of neonatal respiratory morbidity based on fully automated fetal lung ultrasound analysis. *Scientific Reports*, 9(1):1–7, 2019.
- [BCG00] Christophe Biernacki, Gilles Celeux, and Gérard Govaert. Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):719–725, 2000.
- [BCM05] A. Buades, B. Coll, and J.-M. Morel. A non-local algorithm for image denoising. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 60–65, 2005.
- [BF07] Jian Bai and Xiang-Chu Feng. Fractional-order anisotropic diffusion for image denoising. *IEEE Transactions on Image Processing*, 16(10):2492–2502, 2007.
- [BGH20] Katherine G Brown, Debabrata Ghosh, and Kenneth Hoyt. Deep learning of spatiotemporal filtering for fast super-resolution ultrasound imaging. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 67(9):1820–1829, 2020.

- [BGOF⁺19] Michal Byra, Michael Galperin, Haydee Ojeda-Fournier, Linda Olson, Mary O’Boyle, Christopher Comstock, and Michael Andre. Breast mass classification in sonography with transfer learning using a deep convolutional neural network and color conversion. *Medical Physics*, 46(2):746–755, 2019.
- [BLM⁺08] Adrian Basarab, Hervé Liebgott, Fabrice Morestin, Andrej Lyshchik, Tatsuya Higashi, Ryo Asato, and Philippe Delachartre. A method for vector displacement estimation with ultrasound imaging and its application for thyroid nodular disease. *Medical Image Analysis*, 12(3):259–274, 2008.
- [Bos96] Frank Bossen. Anisotropic mesh generation with particles. Technical report, Carnegie-Mellon Univ. Pittsburgh PA Dept. of Computer Science, 1996.
- [BPC⁺11] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, January 2011.
- [BPMJ⁺18] Núria Baños, Alvaro Perez-Moreno, Carla Julià, Clara Murillo-Bravo, David Coronado, Eduard Gratacos, Jan Deprest, and Montse Palacio. Quantitative analysis of cervical texture by ultrasound in mid-pregnancy and association with spontaneous preterm birth. *Ultrasound in Obstetrics & Gynecology*, 51(5):637–643, 2018.
- [BR19] Joshua Batson and Loic Royer. Noise2Self: blind denoising by self-supervision. *arXiv:1901.11365*, 2019.
- [BSD09] Michael Balzer, Thomas Schlömer, and Oliver Deussen. Capacity-constrained point distributions: a variant of Lloyd’s method. *ACM Transactions on Graphics*, 28(3), 2009.
- [BSD18] Biswajit Biswas, Biplab Kanti Sen, and Kashi Nath Dey. Ultrasound medical image deblurring and denoising method using variational model on CUDA. In *Advanced Computing and Systems for Security*, pages 95–108. Springer, 2018.
- [BSGR⁺21] Or Bar-Shira, Ahuva Grubstein, Yael Rapson, Dror Suhami, Eli Atar, Keren Peri-Hanania, Ronnie Rosen, and Yonina C Eldar. Learned super resolution ultrasound for improved breast lesion characterization. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 109–118. Springer, 2021.
- [Bur78] Christoph B Burckhardt. Speckle in ultrasound B-mode scans. *IEEE Transactions on Sonics and Ultrasonics*, 25(1):1–6, 1978.

- [BWWM10] John Bowers, Rui Wang, Li-Yi Wei, and David Maletz. Parallel Poisson disk sampling with spectrum analysis on surfaces. In *ACM Transactions on Graphics*, volume 29, page 166, 2010.
- [CA09] Vincent WS Chan and Sherif Abbas. *Ultrasound Imaging for Regional Anesthesia: a Practical Guide*. Ultrasound for Regional Anesthesia, 2009.
- [CCLM13] Hung-Kuo Chu, Chia-Sheng Chang, Ruen-Rone Lee, and Niloy J Mitra. Halftone QR codes. *ACM Transactions on Graphics*, 32(6):217, 2013.
- [CCS10] Jian-Feng Cai, Emmanuel J Candès, and Zuowei Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.
- [CD06] Alan Kaylor Cline and Inderjit S. Dhillon. *Computation of the singular value decomposition*. CRC Press, 2006.
- [CGB19] Monica Caballero, Jon Ander Gómez, and Aimilia Bantouna. Deep-learning and hpc to boost biomedical applications for health (deephealth). In *32nd International Symposium on Computer-Based Medical Systems*, pages 150–155. IEEE, 2019.
- [CGW⁺13] Jiating Chen, Xiaoyin Ge, Li-Yi Wei, Bin Wang, Yusu Wang, Huamin Wang, Yun Fei, Kang-Lai Qian, Jun-Hai Yong, and Wenping Wang. Bilateral blue noise sampling. *ACM Transactions on Graphics*, 32(6):216, 2013.
- [CHH05] GT Clement, J Huttunen, and K Hynynen. Superresolution ultrasound imaging using back-projected reconstruction. *The Journal of the Acoustical Society of America*, 118(6):3953–3960, 2005.
- [CHP⁺08] Pierrick Coupé, Pierre Hellier, Sylvain Prima, Charles Kervrann, and Christian Barillot. 3D wavelet subbands mixing for image denoising. *International Journal of Biomedical Imaging*, 2008, 2008.
- [CHY⁺21] Jiangang Chen, Chao He, Jintao Yin, Jiawei Li, Xiaoqian Duan, Yucheng Cao, Li Sun, Menghan Hu, Wenfang Li, and Qingli Li. Quantitative analysis and automated lung ultrasound scoring for evaluating COVID-19 pneumonia with neural networks. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 68(7):2507–2515, 2021.
- [CKH⁺18] Woosuk Choi, Mina Kim, Jae HakLee, Jungho Kim, and Jong BeomRa. Deep CNN-based ultrasound super-resolution for high-speed high-resolution B-mode imaging. In *International Ultrasonics Symposium*, pages 1–4. IEEE, 2018.

- [CNP21] Simone Cammarasana, Paolo Nicolardi, and Giuseppe Patanè. A universal deep learning framework for real-time denoising of ultrasound images. *arXiv preprint arXiv:2101.09122*, 2021.
- [CÖMS19] Mine Cüneyitoğlu Özkul, Ünal Erkan Mumcuoğlu, and İbrahim Tanzer Sancı. Single-image Bayesian restoration and multi-image super-resolution restoration for B-mode ultrasound using an accurate system model involving correlated nature of the speckle noise. *Ultrasonic Imaging*, 41(6):368–386, 2019.
- [Coo86] Robert L Cook. Stochastic sampling in Computer Graphics. *ACM Transactions on Graphics*, 5(1):51–72, 1986.
- [CP99] Frank M Candocia and Jose C Principe. Super-resolution of images based on local correlations. *IEEE Transactions on Neural Networks*, 10(2):372–380, 1999.
- [CP11] Vincent Chan and Anahi Perlas. *Basics of Ultrasound Imaging*, pages 13–19. Springer New York, New York, NY, 2011.
- [CP20] Simone Cammarasana and Giuseppe Patanè. Kernel-based sampling of arbitrary data. The Eurographics Association, 2020.
- [CP21] Simone Cammarasana and Giuseppe Patanè. Kernel-based sampling of arbitrary signals. *Computer-Aided Design*, 141:103103, 2021.
- [CSJ⁺10] Heng-Da Cheng, Juan Shan, Wen Ju, Yanhui Guo, and Ling Zhang. Automated breast cancer detection and classification using ultrasound images: a survey. *Pattern Recognition*, 43(1):299–317, 2010.
- [CST00] Nello Cristianini and John Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press, 2000.
- [CYC⁺12] Zhonggui Chen, Zhan Yuan, Yi-King Choi, Ligang Liu, and Wenping Wang. Variational blue noise sampling. *IEEE Transactions on Visualization and Computer Graphics*, 18(10):1784–1796, 2012.
- [CYV00] S Grace Chang, Bin Yu, and Martin Vetterli. Adaptive wavelet thresholding for image denoising and compression. *IEEE Transactions on Image Processing*, 9(9):1532–1546, 2000.
- [CZC⁺18] Zhonggui Chen, Tieyi Zhang, Juan Cao, Yongjie Jessica Zhang, and Cheng Wang. Point cloud resampling using centroidal Voronoi tessellation methods. *Computer-Aided Design*, 102:12–21, 2018.

- [DCC95] David O Draper, J Chris Castel, and Dawn Castel. Rate of temperature increase in human muscle during 1 MHz and 3 MHz continuous ultrasound. *Journal of Orthopaedic & Sports Physical Therapy*, 22(4):142–150, 1995.
- [DCZD06] Arthur L Da Cunha, Jianping Zhou, and Minh N Do. The nonsubsampling contourlet transform: theory, design, and applications. *IEEE Transactions on Image Processing*, 15(10):3089–3101, 2006.
- [Dem97] James W Demmel. *Applied numerical linear algebra*. SIAM, 1997.
- [DFKE06] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising with block-matching and 3D filtering. In *Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning*, volume 6064, page 606414. SPIE, 2006.
- [DFKE09] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Bm3D image denoising with shape-adaptive principal component analysis. In Rémi Gribonval, editor, *Signal Processing with Adaptive Sparse Structured Representations*, 2009.
- [DGA⁺17] Konstantinos Diamantis, Alan H Greenaway, Tom Anderson, Jørgen Arendt Jensen, Paul A Dalgarno, and Vassilis Sboros. Super-resolution axial localization of ultrasound scatter using multi-focal imaging. *IEEE Transactions on Biomedical Engineering*, 65(8):1840–1851, 2017.
- [DGBOD12] Fernando De Goes, Katherine Breeden, Victor Ostromoukhov, and Mathieu Desbrun. Blue noise through optimal transport. *ACM Transactions on Graphics*, 31(6):1–11, 2012.
- [DH06] Daniel Dunbar and Greg Humphreys. A spatial data structure for fast Poisson-disk sample generation. In *ACM Transactions on Graphics*, volume 25, pages 503–508, 2006.
- [DHVOS00] Oliver Deussen, Stefan Hiller, Cornelius Van Overveld, and Thomas Strothotte. Floating points: a method for computing stipple drawings. *Computer Graphics Forum*, 19(3):41–50, 2000.
- [DK90] James Demmel and William Kahan. Accurate singular values of bidiagonal matrices. *SIAM Journal on Scientific and Statistical Computing*, 11(5):873–912, 1990.
- [DLHT14] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European Conference on Computer Vision*, pages 184–199. Springer, 2014.

- [DLHT15] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):295–307, 2015.
- [DO07] Charles Donohue and Victor Ostromoukhov. Fast generation of importance-sampled point sets with associated Delaunay triangulation. In *Proceedings of Graphicon*, pages 125–130. Citeseer, 2007.
- [DP20] Jayashrita Debnath and Michele Parrinello. Gaussian mixture-based enhanced sampling for statics and dynamics. *The Journal of Physical Chemistry Letters*, 11(13):5076–5080, 2020.
- [DSFC⁺13] Tharindu De Silva, Aaron Fenster, Derek W Cool, Lori Gardi, Cesare Romagnoli, Jagath Samarabandu, and Aaron D Ward. 2D-3D rigid registration to compensate for prostate motion during 3D TRUS-guided biopsy. *Medical Physics*, 40(2):022904, 2013.
- [DSL12] Weisheng Dong, Guangming Shi, and Xin Li. Nonlocal image restoration with bilateral variance estimation: a low-rank approach. *IEEE Transactions on Image Processing*, 22(2):700–711, 2012.
- [DW85] Mark AZ Dippé and Erling Henry Wold. Antialiasing through stochastic sampling. *ACM SIGGRAPH Computer Graphics*, 19(3):69–78, 1985.
- [DZSL12] Weisheng Dong, Lei Zhang, Guangming Shi, and Xin Li. Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing*, 22(4):1620–1630, 2012.
- [DZTN21] Jianrui Ding, Shili Zhao, Fenghe Tang, and Chunping Ning. Ultrasound image super-resolution with two-stage zero-shot cycleGAN. In *Journal of Physics: Conference Series*, volume 2031, page 012015. IOP Publishing, 2021.
- [EPP⁺15] Claudia Errico, Juliette Pierre, Sophie Pezet, Yann Desailly, Zsolt Lenkei, Olivier Couture, and Mickael Tanter. Ultrafast ultrasound localization microscopy for deep super-resolution vascular imaging. *Nature*, 527(7579):499–502, 2015.
- [EVDH10] Anders Eriksson and Anton Van Den Hengel. Efficient computation of robust low-rank matrix approximations in the presence of missing data using the L1 norm. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 771–778, 2010.
- [EVW10] Michael A Ellis, Francesco Viola, and William F Walker. Super-resolution image reconstruction using diffuse source models. *Ultrasound in Medicine & Biology*, 36(6):967–977, 2010.

- [Fat11] Raanan Fattal. Blue-noise point sampling using kernel density model. In *ACM Transactions on Graphics*, volume 30, page 48, 2011.
- [FFC⁺20] Jie Feng, Xueliang Feng, Jiantong Chen, Xianghai Cao, Xiangrong Zhang, Licheng Jiao, and Tao Yu. Generative adversarial networks based on collaborative learning and attention mechanism for hyperspectral image classification. *Remote Sensing*, 12(7):1149, 2020.
- [FH60] George Elmer Forsythe and Peter Henrici. The cyclic Jacobi method for computing the principal values of a complex matrix. *Transactions of the American Mathematical Society*, 94(1):1–23, 1960.
- [FH02] AS Fruchter and RN Hook. Drizzle: a method for the linear reconstruction of undersampled images. *Publications of the Astronomical Society of the Pacific*, 114(792):144, 2002.
- [FHHA19] Lin Fu, Luhui Han, Xiangyu Y Hu, and Nikolaus A Adams. An isotropic unstructured mesh generation method based on a fluid relaxation analogy. *Computer Methods in Applied Mechanics and Engineering*, 350:396–431, 2019.
- [FJZ⁺21] Xiyu Fu, Sen Jia, Lina Zhuang, Meng Xu, Jun Zhou, and Qingquan Li. Hyperspectral anomaly detection via deep plug-and-play denoising CNN regularization. *IEEE Transactions on Geoscience and Remote Sensing*, 59(11):9553–9568, 2021.
- [FM16] Amira Hadj Fredj and Jihene Malek. Real time ultrasound image denoising using NVIDIA CUDA. In *2nd International Conference on Advanced Technologies for Signal and Image Processing*, pages 136–140. IEEE, 2016.
- [FP04] Jean-Philippe Farrugia and Bernard Péroche. A progressive rendering algorithm using an adaptive perceptually based image metric. In *Computer Graphics Forum*, volume 23, pages 605–614, 2004.
- [FSSH82] Victor S Frost, Josephine Abbott Stiles, K Sam Shanmugan, and Julian C Holtzman. A model for radar images and its application to adaptive digital filtering of multiplicative noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(2):157–166, 1982.
- [FT16] Björn Fabritius and G Tabor. Improving the quality of finite volume meshes through genetic optimisation. *Engineering with Computers*, 32(3):425–440, 2016.
- [FZ20] Yingying Fang and Tiejong Zeng. Learning deep edge prior for image denoising. *Computer Vision and Image Understanding*, 200:103044, 2020.

- [FZI⁺17] Josquin Foiret, Hua Zhang, Tali Ilovitsh, Lisa Mahakian, Sarah Tam, and Katherine W Ferrara. Ultrasound localization microscopy to image and assess microvasculature in a rat kidney. *Scientific Reports*, 7(1):1–12, 2017.
- [GAD12] Ryan C Gessner, Stephen R Aylward, and Paul A Dayton. Mapping microvasculature with acoustic angiography yields quantifiable differences between healthy and tumor-bearing tissue volumes in a rodent model. *Radiology*, 264(3):733, 2012.
- [GBL⁺21] Krishan Gupta, Kirti Balyan, Bhumika Lamba, Manju Puri, Debarka Sen Gupta, and Manisha Kumar. Ultrasound placental image texture analysis using artificial intelligence to predict hypertension in pregnancy. *The Journal of Maternal-Fetal & Neonatal Medicine*, pages 1–8, 2021.
- [GE95] Ming Gu and Stanley C Eisenstat. A divide-and-conquer algorithm for the bidiagonal SVD. *SIAM Journal on Matrix Analysis and Applications*, 16(1):79–92, 1995.
- [GEB16] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016.
- [GG84] Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):721–741, 1984.
- [Gib08] Bernard Gibaud. The DICOM standard: a brief overview. *Molecular Imaging: Computer Reconstruction and Practice*, pages 229–238, 2008.
- [GJB03] KT Gribbon, CT Johnston, and Donald G Bailey. A real-time FPGA implementation of a barrel distortion correction algorithm with bilinear interpolation. In *Image and Vision Computing New Zealand*, pages 408–413, 2003.
- [GK19] Amit Garg and Vineet Khandelwal. Despeckling of medical ultrasound images using fast bilateral filter and neighshrinksure filter in wavelet domain. In *Advances in Signal Processing and Communication*, pages 271–280. Springer, 2019.
- [GLR⁺16] Peng Gu, Won-Mean Lee, Marilyn A Roubidoux, Jie Yuan, Xueding Wang, and Paul L Carson. Automated 3D ultrasound image segmentation to aid breast cancer image interpretation. *Ultrasonics*, 65:51–58, 2016.
- [GRG06] Hayit Greenspan, Amit Ruf, and Jacob Goldberger. Constrained Gaussian mixture model framework for automatic segmentation of MR brain images. *IEEE Transactions on Medical Imaging*, 25(9):1233–1245, 2006.

- [GVV⁺20] Leah A Groves, Blake VanBerlo, Natan Veinberg, Abdulrahman Alboog, Terry M Peters, and Elvis Chen. Automatic segmentation of the carotid artery and internal jugular vein from 2D ultrasound images for 3D vascular reconstruction. *International Journal of Computer Assisted Radiology and Surgery*, 15(11):1835–1846, 2020.
- [GZZF14] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2862–2869, 2014.
- [Han14] Uwe D Hanebeck. Kernel-based deterministic blue-noise sampling of arbitrary probability density functions. In *Conference on Information Sciences and Systems*, pages 1–6. IEEE, 2014.
- [HCJB⁺18] Sevan Harput, Kirsten Christensen-Jeffries, Jemma Brown, Yuanwei Li, Katherine J Williams, Alun H Davies, Robert J Eckersley, Christopher Dunsby, and Meng-Xing Tang. Two-stage motion correction for super-resolution ultrasound imaging in human lower limb. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 65(5):803–814, 2018.
- [Hes58] Magnus R Hestenes. Inversion of matrices by biorthogonalization and related results. *Journal of the Society for Industrial and Applied Mathematics*, 6(1):51–90, 1958.
- [HLZ17] Qinghua Huang, Yaozhong Luo, and Qiangzhi Zhang. Breast ultrasound image segmentation: a survey. *International Journal of Computer Assisted Radiology and Surgery*, 12(3):493–507, 2017.
- [HNTL18] Zhanxuan Hu, Feiping Nie, Lai Tian, and Xuelong Li. A comprehensive survey for low rank regularization. In *arXiv: 1808.04521*. 2018.
- [Hou58] Alston S Householder. Unitary triangularization of a nonsymmetric matrix. *Journal of the ACM*, 5(4):339–342, 1958.
- [HS81] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185–203, 1981.
- [HSA15] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2015.
- [HSD13] Daniel Heck, Thomas Schlömer, and Oliver Deussen. Blue noise sampling with controlled aliasing. *ACM Transactions on Graphics*, 32(3):1–12, 2013.

- [HW94] Stefan W Hell and Jan Wichmann. Breaking the diffraction resolution limit by stimulated emission: stimulated-emission-depletion fluorescence microscopy. *Optics letters*, 19(11):780–782, 1994.
- [ICM21] Ademola E Ilesanmi, Utairat Chaumrattanakul, and Stanislav S Makhanov. Methods for the segmentation and classification of breast ultrasound images: a review. *Journal of Ultrasound*, 24(4):367–382, 2021.
- [IKM08] Dimitris K Iakovidis, Eystratios G Keramidas, and Dimitris Maroulis. Fuzzy local binary patterns for ultrasound texture characterization. In *International Conference Image Analysis and Recognition*, pages 750–759. Springer, 2008.
- [IZZE17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1125–1134, 2017.
- [JGB⁺20] Pankaj K Jain, Saurabh Gupta, Arnav Bhavsar, Aditya Nigam, and Neeraj Sharma. Localization of common carotid artery transverse section in B-mode ultrasound images using faster RCNN: a deep learning approach. *Medical & Biological Engineering & Computing*, 58(3):471–482, 2020.
- [JLSX10] Hui Ji, Chaoqiang Liu, Zuowei Shen, and Yuhong Xu. Robust video denoising using low rank matrix completion. In *Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1791–1798. IEEE, 2010.
- [JTJT20] Manuel Jiménez, Mercedes Torres Torres, Robert John, and Isaac Triguero. Galaxy image classification based on citizen science data: a comparative study. *IEEE Access*, 8:47232–47246, 2020.
- [KAJ⁺20] Donya Khaledyan, Abdollah Amirany, Kian Jafari, Mohammad Hossein Moaiyeri, Abolfazl Zargari Khuzani, and Najmeh Mashhadi. Low-cost implementation of bilinear and bicubic image interpolation for real-time image super-resolution. In *Global Humanitarian Technology Conference*, pages 1–5. IEEE, 2020.
- [KAR18] Parviz Khavari, Amir Asif, and Hassan Rivaz. Non-local super resolution in ultrasound imaging. In *20th International Workshop on Multimedia Signal Processing*, pages 1–6. IEEE, 2018.
- [KBC07] Charles Kervrann, Jérôme Boulanger, and Pierrick Coupé. Bayesian non-local means filter, image redundancy and adaptive dictionaries for noise removal. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 520–532. Springer, 2007.

- [KBJ19] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2Void-learning denoising from single noisy images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2129–2137, 2019.
- [KBSA00] Gil Kovalski, Rafael Beyar, Rona Shofti, and Haim Azhari. Three-dimensional automatic quantitative analysis of intravascular ultrasound images. *Ultrasound in Medicine & Biology*, 26(4):527–537, 2000.
- [KCODL06] Johannes Kopf, Daniel Cohen-Or, Oliver Deussen, and Dani Lischinski. Recursive wang tiles for real-time blue noise. In *ACM SIGGRAPH*, pages 509–518. 2006.
- [Key81] Robert Keys. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal processing*, 29(6):1153–1160, 1981.
- [KH01] Alexander Keller and Wolfgang Heidrich. Interleaved sampling. In *Rendering Techniques 2001*, pages 269–276. Springer, 2001.
- [KK19] Shizuo Kaji and Satoshi Kida. Overview of image-to-image translation by use of deep neural networks: denoising, super-resolution, modality conversion, and reconstruction in medical imaging. *Radiological Physics and Technology*, 12(3):235–248, 2019.
- [KL80] Virginia Klema and Alan Laub. The singular value decomposition: its computation and some applications. *IEEE Transactions on Automatic Control*, 25(2):164–176, 1980.
- [KLR16] Gitta Kutyniok, Wang-Q Lim, and Rafael Reisenhofer. Shearlab 3D: faithful digital shearlet transforms based on compactly supported shearlets. *ACM Transactions on Mathematical Software*, 42(1):1–42, 2016.
- [KMM⁺03] A Kong, P McCullagh, X-L Meng, D Nicolae, and Z Tan. A theory of statistical models for Monte Carlo integration. *Journal of the Royal Statistical Society: Series B*, 65(3):585–604, 2003.
- [Kog55] EG Kogbetliantz. Solution of linear equations by diagonalization of coefficients matrix. *Quarterly of Applied Mathematics*, 13(2):123–132, 1955.
- [KS13] Nolan Kurtz and Junho Song. Cross-entropy-based adaptive importance sampling using Gaussian mixture. *Structural Safety*, 42:35–44, 2013.
- [KSH17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

- [KSSC85] Darwin T Kuan, Alexander A Sawchuk, Timothy C Strand, and Pierre Chavel. Adaptive noise smoothing filter for images with signal-dependent noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(2):165–177, 1985.
- [LA01] Daniel L Lau and Gonzalo R Arce. *Modern digital halftoning*. CRC Press, 2001.
- [LBK⁺18] Jeong Hoon Lee, Jung Hwan Baek, Ju Han Kim, Woo Hyun Shim, Sae Rom Chung, Young Jun Choi, and Jeong Hyun Lee. Deep learning–based computer-aided diagnosis system for localization and diagnosis of metastatic lymph nodes on ultrasound: a pilot study. *Thyroid*, 28(10):1332–1338, 2018.
- [LCW20] Hui Liu, Wen Chu, and Hua Wang. Automatic segmentation algorithm of ultrasound heart image based on convolutional neural network and image saliency. *IEEE Access*, 8:104445–104457, 2020.
- [Lee80] J. Lee. Digital image enhancement and noise filtering by use of local statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(2):165–168, 1980.
- [LFL⁺20] Xiaowen Liang, Jinghui Fang, Haoming Li, Xin Yang, Dong Ni, Fengyi Zeng, and Zhiyi Chen. CR-unet-based ultrasonic follicle monitoring to reduce diameter variability and generate area automatically as a novel biomarker for follicular maturity. *Ultrasound in Medicine & Biology*, 46(11):3125–3134, 2020.
- [LFL⁺21] Xin Liu, Yiting Fan, Shuang Li, Meixiang Chen, Ming Li, William Kongto Hau, Heye Zhang, Lin Xu, and Alex Pui-Wai Lee. Deep learning-based automated left ventricular ejection fraction assessment using 2-D echocardiography. *American Journal of Physiology-Heart and Circulatory Physiology*, 321(2):H390–H399, 2021.
- [LGL⁺19] Tianjiao Liu, Qianqian Guo, Chunfeng Lian, Xuhua Ren, Shujun Liang, Jing Yu, Lijuan Niu, Weidong Sun, and Dinggang Shen. Automated detection and classification of thyroid nodules in ultrasound images using clinical-knowledge-guided convolutional neural networks. *Medical Image Analysis*, 58:101555, 2019.
- [Li06] Xin Li. Edge-directed error diffusion halftoning. *Signal Processing Letters*, 13(11):688–690, 2006.
- [Lin04] Fredrik Lingvall. A method of improving overall resolution in ultrasonic array imaging using spatio-temporal deconvolution. *Ultrasonics*, 42(1-9):961–968, 2004.

- [LKO06] Roberto Lavarello, Farzad Kamalabadi, and William D O'Brien. A regularized inverse approach to ultrasonic pulse-echo imaging. *IEEE Transactions on Medical Imaging*, 25(6):712–722, 2006.
- [LL18] Jingfeng Lu and Wanyu Liu. Unsupervised super-resolution framework for medical ultrasound images using dilated convolutional neural networks. In *3rd International Conference on Image, Vision and Computing*, pages 739–744. IEEE, 2018.
- [LLH⁺21] Heng Liu, Jianyong Liu, Shudong Hou, Tao Tao, and Jungong Han. Perception consistency ultrasound image super-resolution via self-supervised CycleGAN. *Neural Computing and Applications*, pages 1–11, 2021.
- [Llo82] Stuart Lloyd. Least-squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.
- [LMH⁺18] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2Noise: learning image restoration without clean data. *arXiv:1803.04189*, 2018.
- [LSK⁺17] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 136–144, 2017.
- [LTF13] Olivier Laligant, Frédéric Truchetet, and Eric Fauvet. Noise estimation from digital step-model signal. *IEEE Transactions on Image Processing*, 22(12):5158–5167, 2013.
- [LTH⁺17] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4681–4690, 2017.
- [LWY⁺19] Shengfeng Liu, Yi Wang, Xin Yang, Baiying Lei, Li Liu, Shawn Xiang Li, Dong Ni, and Tianfu Wang. Deep learning in medical ultrasound analysis: a review. *Engineering*, 5(2):261–275, 2019.
- [LZC⁺17] Xin Liu, He Zhang, Yiu-ming Cheung, Xinge You, and Yuan Yan Tang. Efficient single image dehazing and denoising: an efficient multi-scale correlated wavelet approach. *Computer Vision and Image Understanding*, 162:23–33, 2017.

- [LZT⁺22] Xingtao Lin, Xiaogen Zhou, Tong Tong, Xingqing Nie, Luoyan Wang, Haonan Zheng, Jing Li, Ensheng Xue, Shun Chen, Meijuan Zheng, et al. A super-resolution guided network for improving automated thyroid nodule segmentation. *Computer Methods and Programs in Biomedicine*, 227:107186, 2022.
- [Mac67] James MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297, 1967.
- [MBFE12] Matteo Maggioni, Giacomo Boracchi, Alessandro Foi, and Karen Egiazarian. Video denoising, deblocking, and enhancement through separable 4-D nonlocal spatiotemporal transforms. *IEEE Transactions on Image Processing*, 21(9):3952–3966, 2012.
- [MBK12] Renaud Morin, Adrian Basarab, and Denis Kouamé. Alternating direction method of multipliers framework for super-resolution in ultrasound imaging. In *International Symposium on Biomedical Imaging*, pages 1595–1598. IEEE, 2012.
- [MBP⁺09] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Non-local sparse models for image restoration. In *IEEE International Conference on Computer Vision*, pages 2272–2279, 2009.
- [MBPK12] Renaud Morin, Adrian Basarab, Marie Ploquin, and Denis Kouamé. Post-processing multiple-frame super-resolution in ultrasound imaging. In *Medical Imaging: Ultrasonic Imaging, Tomography, and Therapy*, volume 8320, pages 433–440. SPIE, 2012.
- [MCMB⁺10] José V Manjón, Pierrick Coupé, Luis Martí-Bonmatí, D Louis Collins, and Montserrat Robles. Adaptive non-local means denoising of MR images with spatially varying noise levels. *Journal of Magnetic Resonance Imaging*, 31(1):192–203, 2010.
- [MCMM09] Cathy Maugis, Gilles Celeux, and Marie-Laure Martin-Magniette. Variable selection for clustering with Gaussian mixture models. *Biometrics*, 65(3):701–709, 2009.
- [MEO13] Bradley A Malin, Khaled El Emam, and Christine M O’Keefe. Biomedical data privacy: problems, perspectives, and recent advances. *Journal of the American Medical Informatics Association*, 20(1):2–6, 2013.
- [Mér11] Quentin Mérigot. A multiscale approach to optimal transport. In *Computer Graphics Forum*, volume 30, pages 1583–1592. Wiley Online Library, 2011.

- [Mit87] Don P Mitchell. Generating antialiased images at low sampling densities. *ACM SIGGRAPH Computer Graphics*, 21(4):65–72, 1987.
- [Mit19] Brent Mittelstadt. The ethics of biomedical ‘big data’ analytics. *Philosophy & Technology*, 32(1):17–21, 2019.
- [MKEF12] Matteo Maggioni, Vladimir Katkovnik, Karen Egiazarian, and Alessandro Foi. Nonlocal transform-domain filter for volumetric data denoising and reconstruction. *IEEE Transactions on Image Processing*, 22(1):119–133, 2012.
- [MKKY18] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.
- [MKRM99] M Kivanc Mihcak, Igor Kozintsev, Kannan Ramchandran, and Pierre Moulin. Low-complexity image denoising based on statistical modeling of wavelet coefficients. *Signal Processing Letters*, 6(12):300–303, 1999.
- [MMP⁺14] Gopinath Mahale, Hamsika Mahale, Rajesh Babu Parimi, SK Nandy, and S Bhattacharya. Hardware architecture of bi-cubic convolution interpolation for real-time image scaling. In *International Conference on Field-Programmable Technology*, pages 264–267. IEEE, 2014.
- [MNB13] Arian Maleki, Manjari Narayan, and Richard G Baraniuk. Anisotropic non-local means denoising. *Applied and Computational Harmonic Analysis*, 35(3):452–482, 2013.
- [MU19] Michael T. McCann and Michael Unser. Biomedical image reconstruction: from the foundations to deep neural networks. *Foundations and Trends in Signal Processing*, 13(3):283–359, 2019.
- [MXZ17] Liyan Ma, Li Xu, and Tiejong Zeng. Low rank prior and total variation regularization for image deblurring. *Journal of Scientific Computing*, 70(3):1336–1357, 2017.
- [MZ16] Liyan Ma and Tiejong Zeng. Image deblurring via total variation based structured sparse model selection. *Journal of Scientific Computing*, 67(1):1–19, 2016.
- [MZY20] Fuyuan Mei, Dong Zhang, and Yan Yang. Improved non-local self-similarity measures for effective speckle noise reduction in ultrasound images. *Computer Methods and Programs in Biomedicine*, 196:105670, 2020.

- [NEK⁺20] Saman Nikeghbalian, Ahad Eshraghian, Kouros Kazemi, Alireza Shamsaeifar, Masood Hosseinzadeh, Bitra Geramizadeh, and Seyed Ali Malek-Hosseini. Liver transplantation for high-grade primary hepatic neuroendocrine tumor with diffuse liver metastasis. *Journal of Gastrointestinal Cancer*, 51(1):304–306, 2020.
- [NWX10] Michael K Ng, Pierre Weiss, and Xiaoming Yuan. Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods. *Journal on Scientific Computing*, 32(5):2710–2736, 2010.
- [ÖAG10] A Cengiz Öztireli, Marc Alexa, and Markus Gross. Spectral sampling of manifolds. *ACM Transactions on Graphics*, 29(6):168, 2010.
- [OH95] Victor Ostromoukhov and Roger D Hersch. Artistic screening. Technical report, ACM, 1995.
- [OHG⁺20] David Ouyang, Bryan He, Amirata Ghorbani, Neal Yuan, Joseph Ebinger, Curtis P Langlotz, Paul A Heidenreich, Robert A Harrington, David H Liang, Euan A Ashley, and James Y. Zou. Video-based AI for beat-to-beat assessment of cardiac function. *Nature*, 580(7802):252–256, 2020.
- [Pat15] Giuseppe Patané. Diffusive smoothing of 3D segmented medical data. *Journal of Advanced Research*, 6(3):425–431, 2015.
- [PBA⁺19] Trupesh R. Patel, Sandeep Bodduluri, Thomas Anthony, William S. Monroe, Pravinkumar G. Kandhare, John-Paul Robinson, Arie Nakhmani, Chengcui Zhang, Surya P. Bhatt, and Purushotham V. Bangalore. Performance characterization of single and multi GPU training of U-Net architecture for medical image segmentation tasks. In *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (Learning)*. ACM, 2019.
- [PBC⁺20] Lois Paulin, Nicolas Bonneel, David Coeurjolly, Jean-Claude Iehl, Antoine Webanck, Mathieu Desbrun, and Victor Ostromoukhov. Sliced optimal transport sampling. *ACM Transactions on Graphics*, 39, 2020.
- [PC20] Witold Pedrycz and Shyi-Ming Chen. *Deep learning: concepts and architectures*. Springer, 2020.
- [PE14] Tomer Peleg and Michael Elad. A statistical prediction model based on sparse representations for single image super-resolution. *IEEE Transactions on Image Processing*, 23(6):2569–2582, 2014.

- [PFFG98] Sumanta N Pattanaik, James A Ferwerda, Mark D Fairchild, and Donald P Greenberg. A multiscale model of adaptation and spatial vision for realistic image display. In *Proceedings of Conference on Computer Graphics and Interactive Techniques*, pages 287–298. Citeseer, 1998.
- [Pia08] Oleg S Pinykh. *DICOM Security*. Springer, 2008.
- [PJH16] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: from theory to implementation*. Morgan Kaufmann, 2016.
- [PKKG03] Mark Pauly, Richard Keiser, Leif P Kobbelt, and Markus Gross. Shape modeling with point-sampled geometry. In *ACM Transactions on Graphics*, volume 22, pages 641–650, 2003.
- [PLLZ21] Bin Pu, Kenli Li, Shengli Li, and Ningbo Zhu. Automatic fetal ultrasound standard plane recognition based on deep learning and IIoT. *IEEE Transactions on Industrial Informatics*, 17(11):7771–7780, 2021.
- [PM90] Pietro Perona and Jitendra Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.
- [PPAV11] Sara Parrilli, Mariana Poderico, Cesario Vincenzo Angelino, and Luisa Verdoliva. A nonlocal sar image denoising algorithm based on LLMMSE wavelet shrinkage. *IEEE Transactions on Geoscience and Remote Sensing*, 50(2):606–616, 2011.
- [PPBN05] M Puri, KM Patil, V Balasubramanian, and VB Narayanamurthy. Texture analysis of foot sole soft tissue images in diabetic neuropathy using wavelet transform. *Medical & Biological Engineering & Computing*, 43(6):756–763, 2005.
- [PQW⁺08] Wai-Man Pang, Yingge Qu, Tien-Tsin Wong, Daniel Cohen-Or, and Pheng-Ann Heng. Structure-aware halftoning. In *ACM Transactions on Graphics*, volume 27, page 89, 2008.
- [PS04] Per-Olof Persson and Gilbert Strang. A simple mesh generator in MATLAB. *SIAM Review*, 46(2):329–345, 2004.
- [PSWS03] Javier Portilla, Vasily Strela, Martin J Wainwright, and Eero P Simoncelli. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11):1338–1351, 2003.

- [PXdFABCH11] Fernanda Palhano Xavier de Fontes, Guillermo Andrade Barroso, Pierrick Coupé, and Pierre Hellier. Real time ultrasound image denoising. *Journal of Real-time Image Processing*, 6(1):15–22, 2011.
- [QCHC17] Hongxing Qin, Yi Chen, Jinlong He, and Baoquan Chen. Wasserstein blue noise sampling. *ACM Transactions on Graphics*, 36(5):1–13, 2017.
- [RB10] Adrian E Raftery and Le Bao. Estimating and projecting trends in HIV/AIDS generalized epidemics using incremental mixture importance sampling. *Biometrics*, 66(4):1162–1173, 2010.
- [RFB15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.
- [RFG⁺18] U Raghavendra, Hamido Fujita, Anjan Gudigar, Ranjan Shetty, Krishnananda Nayak, Umesh Pai, Jyothi Samanth, and U Rajendra Acharya. Automated technique for coronary artery disease characterization and classification using DD-DTDWT in ultrasound images. *Biomedical Signal Processing and Control*, 40:324–334, 2018.
- [RGBP18] Prasun Roy, Subhankar Ghosh, Saumik Bhattacharya, and Umapada Pal. Effects of degradations on deep neural network architectures. *arXiv:1807.10108*, 2018.
- [RGÇ⁺18] Nishant Ravikumar, Ali Gooya, Serkan Çimen, Alejandro F Frangi, and Zeike A Taylor. Group-wise similarity registration of point sets using Student’s t-mixture model for statistical shape models. *Medical Image Analysis*, 44:156–176, 2018.
- [RHA07] Gustavo K Rohde, DM Healy, and Akram Aldroubi. Sampling and reconstruction for biomedical image registration. In *Conference Record of the Forty-First Asilomar Conference on Signals, Systems and Computers*, pages 220–223. IEEE, 2007.
- [RHC⁺19] Maosong Ran, Jinrong Hu, Yang Chen, Hu Chen, Huaqiang Sun, Jiliu Zhou, and Yi Zhang. Denoising of 3D magnetic resonance images using a residual encoder–decoder Wasserstein generative adversarial network. *Medical Image Analysis*, 55:165–180, 2019.
- [RMC15] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv:1511.06434*, 2015.

- [Rot82] Scott D Roth. Ray casting for modeling solids. *Computer Graphics and Image Processing*, 18(2):109–144, 1982.
- [RR20] Nathanaël Carraz Rakotonirina and Andry Rasoanaivo. ESRGAN+: further improving enhanced super-resolution generative adversarial network. In *International Conference on Acoustics, Speech and Signal Processing*, pages 3637–3641. IEEE, 2020.
- [RRB12] Ajit Rajwade, Anand Rangarajan, and Arunava Banerjee. Image denoising using the higher order singular value decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4):849–862, 2012.
- [Rud16] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.
- [Rus03] Fabrizio Russo. A method for estimation and filtering of Gaussian noise in images. *IEEE Transactions on Instrumentation and Measurement*, 52(4):1148–1154, 2003.
- [RV15] Tim Roughgarden and Gregory Valiant. Cs168: The modern algorithmic toolbox lecture# 9: the singular value decomposition (SVD) and low-rank matrix approximations. <http://theory.stanford.edu/~tim/s15/l/19.pdf>, 2015. Accessed: 2022-09-18.
- [SC21] Jae Woong Soh and Nam Ik Cho. Deep universal blind image denoising. In *25th International Conference on Pattern Recognition*, pages 747–754. IEEE, 2021.
- [SCD02] Jean-Luc Starck, Emmanuel J Candès, and David L Donoho. The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11(6):670–684, 2002.
- [SCYY11] Hayden So, Junying Chen, Billy Yiu, and Alfred Yu. Medical ultrasound imaging: to GPU or not to GPU? *Micro*, 31(5):54–65, 2011.
- [SD07] Ting Su and Jennifer G Dy. In search of deterministic methods for initializing k-means and Gaussian mixture clustering. *Intelligent Data Analysis*, 11(4):319–338, 2007.
- [Sec02] Adrian Secord. Weighted Voronoi stippling. In *Proceedings of the Symposium on non-photorealistic animation and rendering*, pages 37–43, 2002.
- [SEDP19] Danai E Soulioti, David Espíndola, Paul A Dayton, and Gianmarco F Pinton. Super-resolution imaging through the human skull. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 67(1):25–36, 2019.

- [SG95] Kenji Shimada and David C Gossard. Bubble mesh: automated triangular meshing of non-manifold geometry by sphere packing. In *Proceedings of the ACM Symposium on Solid Modeling and Applications*, pages 409–419, 1995.
- [SGBW10] Christian Schmaltz, Pascal Gwosdek, Andrés Bruhn, and Joachim Weickert. Electrostatic halftoning. *Computer Graphics Forum*, 29(8):2313–2327, 2010.
- [SHA03] Volker Strumpfen, Henry Hoffmann, and Anant Agarwal. A stream algorithm for the SVD. *Technical Memo, MIT-LCS-TM-641*, pages 1–29, 2003.
- [She04] Simon J Sheather. Density estimation. *Statistical science*, pages 588–597, 2004.
- [SJ03] Nathan Srebro and Tommi Jaakkola. Weighted low-rank approximations. In *Proceedings of the 20th International Conference on Machine Learning*, pages 720–727, 2003.
- [SJ21] Revathy Sivanandan and J Jayakumari. A new CNN architecture for efficient classification of ultrasound breast tumor images with activation map clustering based prediction validation. *Medical & Biological Engineering & Computing*, 59(4):957–968, 2021.
- [SL03] Ivan W Selesnick and Ke Yong Li. Video denoising using 2D and 3D dual-tree complex wavelet transforms. In *Wavelets: Applications in Signal and Image Processing X*, volume 5207, pages 607–618. International Society for Optics and Photonics, 2003.
- [SLW⁺20] Lan Shen, Min Liu, Chao Wang, Changhao Guo, Erik Meijering, and Yaonan Wang. Efficient 3D junction detection in biomedical images based on a circular sampling model and reverse mapping. *Journal of Biomedical and Health Informatics*, 25(5):1612–1623, 2020.
- [SM05] Christopher Lee Siström and Niccie L McKay. Costs, charges, and revenues for hospital diagnostic imaging procedures: differences by modality and hospital characteristics. *Journal of the American College of Radiology*, 2(6):511–519, 2005.
- [SP04] Miguel Sainz and Renato Pajarola. Point-based rendering techniques. *Computers & Graphics*, 28(6):869–879, 2004.
- [SPP15] Jordi Salvador and Eduardo Perez-Pellitero. Naive Bayes super-resolution forest. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 325–333, 2015.

- [SS21] Mukesh Soni and Dileep Kumar Singh. Blockchain-based security & privacy for biomedical and healthcare information exchange systems. *Materials Today: Proceedings*, 2021.
- [SSGL07] Amit Shrivastava, Monika Shinde, SS Gornale, and Pratap Lawande. An approach-effect of an exponential distribution on different medical images. *International Journal of Computer Science and Network Security*, 7(9):235, 2007.
- [SZ14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014.
- [SZA⁺21] Scott Schoen, Zhigen Zhao, Ashley Alva, Chengwu Huang, Shigao Chen, and Costas Arvanitis. Morphological reconstruction improves microvessel mapping in super-resolution ultrasound. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 68(6):2141–2149, 2021.
- [TAF15] Afonso M Teodoro, Mariana SC Almeida, and Mário AT Figueiredo. Single-frame image denoising and inpainting using Gaussian mixtures. In *International Conference on Pattern Recognition Applications and Methods*, pages 283–288, 2015.
- [TB20] Hakan Temiz and Hasan S Bilge. Super resolution of B-mode ultrasound images with deep learning. *Access*, 8:78808–78820, 2020.
- [TDSVG13] Radu Timofte, Vincent De Smet, and Luc Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1920–1927, 2013.
- [TDSVG14] Radu Timofte, Vincent De Smet, and Luc Van Gool. A+: adjusted anchored neighborhood regression for fast super-resolution. In *Asian Conference on Computer Vision*, pages 111–126. Springer, 2014.
- [TFZ⁺20] Chunwei Tian, Lunke Fei, Wenxian Zheng, Yong Xu, Wangmeng Zuo, and Chia-Wen Lin. Deep learning on image denoising: an overview. *Neural Networks*, 131:251–275, 2020.
- [TGS⁺10] Jinshan Tang, Shengwen Guo, Qingling Sun, Youping Deng, and Dongfeng Zhou. Speckle reducing bilateral filter for cattle follicle segmentation. *BMC genomics*, 11(2):1–9, 2010.
- [TJ04] Torfinn Taxt and Radovan Jirik. Superresolution of ultrasound images using the first and second harmonic signal. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 51(2):163–175, 2004.

- [TMDCSN21] Hamidreza Taleghamar, Hadi Moghadas-Dastjerdi, Gregory J Czarnota, and Ali Sadeghi-Naini. Characterizing intra-tumor regions on quantitative ultrasound parametric images to predict breast cancer response to chemotherapy at pre-treatment. *Scientific Reports*, 11(1):1–13, 2021.
- [TV91] Demetri Terzopoulos and Manuela Vasilescu. Sampling and reconstruction with adaptive meshes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 70–75, 1991.
- [TZ12] H Joel Trussell and R Zhang. The dominance of poisson noise in color digital cameras. In *19th International Conference on Image Processing*, pages 329–332. IEEE, 2012.
- [urla] Cineca marconi100. <https://www.top500.org/system/179845/>. Accessed: 2022-04-20.
- [urlb] Esaote mylab-9,. <https://www.esaote.com/ultrasound/ultrasound-systems/p/mylab-9/>. Accessed: 2022-02-11.
- [urlc] Philips epiq elite,. https://www.philips.it/c-dam/b2bhc/it/events/siumb/Epiq_Elite_GISS_6.0_452299156181_LR200-vb.pdf. Accessed: 2022-02-11.
- [urld] Philips xres pro,. <https://www.usa.philips.com/healthcare/education-resources/technologies/ultrasound/xres>. Accessed: 2022-02-11.
- [VEW07] Francesco Viola, Michael A Ellis, and William F Walker. Time-domain optimized near-field estimator for ultrasound imaging: initial development and results. *IEEE Transactions on Medical Imaging*, 27(1):99–110, 2007.
- [VP17] Rajiv Verma and Rajoo Pandey. Adaptive selection of search region for nlm based image denoising. *Optik*, 147:151–162, 2017.
- [vSCE20] Ruud J. G. van Sloun, Regev Cohen, and Yonina C. Eldar. Deep learning in ultrasound imaging. *Proceedings of the IEEE*, 108(1):11–29, 2020.
- [VSSB⁺19] Ruud JG Van Sloun, Oren Solomon, Matthew Bruce, Zin Z Khaing, Yonina C Eldar, and Massimo Misch. Deep learning for super-resolution vascular ultrasound imaging. In *International Conference on Acoustics, Speech and Signal Processing*, pages 1055–1059. IEEE, 2019.
- [WCE07] Kenric B White, David Cline, and Parris K Egbert. Poisson disk point sets by hierarchical dart throwing. In *IEEE Symposium on Interactive Ray Tracing*, pages 129–132, 2007.

- [WCH20] Zhihao Wang, Jian Chen, and Steven CH Hoi. Deep learning for image super-resolution: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3365–3387, 2020.
- [WDW⁺20] Mengwan Wei, Yongzhao Du, Xiuming Wu, Qichen Su, Jianqing Zhu, Lixin Zheng, Guorong Lv, and Jiafu Zhuang. A benign and malignant breast tumor classification method via efficiently combining texture and morphological features on ultrasound images. *Computational and Mathematical Methods in Medicine*, 2020, 2020.
- [Web97] Allan G Weber. The USC-SIPI image database version 5. *USC-SIPI Report*, 315(1), 1997.
- [WEG87] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1-3):37–52, 1987.
- [WHA⁺21] Wei Wei, Xu Haishan, Julian Alpers, Marko Rak, and Christian Hansen. A deep learning approach for 2D ultrasound and 3D CT/MR image registration in liver tumor ablation. *Computer Methods and Programs in Biomedicine*, 206:106117, 2021.
- [WJJ⁺16] CM Wendl, M Janke, W Jung, C Stroszczysnski, and EM Jung. Contrast-enhanced ultrasound with perfusion analysis for the identification of malignant and benign tumours of the thyroid gland. *Clinical Hemorheology and Microcirculation*, 63(2):113–121, 2016.
- [WLJ⁺20] Tingting Wu, Wei Li, Shilong Jia, Yiqiu Dong, and Tiejong Zeng. Deep multi-level wavelet-CNN denoiser prior for restoring blurred image with Cauchy noise. *Signal Processing Letters*, 27:1635–1639, 2020.
- [WLL16] Lanjun Wan, Kenli Li, and Keqin Li. A novel cooperative accelerated parallel two-list algorithm for solving the subset-sum problem on a hybrid CPU–GPU cluster. *Journal of Parallel and Distributed Computing*, 97:112–123, 2016.
- [WRB18] Bihan Wen, Saiprasad Ravishankar, and Yoram Bresler. Vidosat: high-dimensional sparsifying transform learning for online video denoising. *IEEE Transactions on Image processing*, 28(4):1691–1704, 2018.
- [WYW⁺18] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. ESRGAN: enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision*, 2018.

- [WZH⁺21] Yuanyuan Wang, Yao Zhang, Qiong He, Hongen Liao, and Jianwen Luo. Quantitative analysis of pleural line and b-lines in lung ultrasound images for severity assessment of COVID-19 pneumonia. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 69(1):73–83, 2021.
- [XLL⁺16] Zhongcheng Xin, Guiting Lin, Hongen Lei, Tom F Lue, and Yinglu Guo. Clinical applications of low-intensity pulsed ultrasound and its potential role in urology. *Translational Andrology and Urology*, 5(2):255, 2016.
- [XYJ17] Shaoping Xu, Xiaohui Yang, and Shunliang Jiang. A fast nonlocally centralized sparse representation algorithm for image denoising. *Signal Processing*, 131:99–112, 2017.
- [XZZ18] Jun Xu, Lei Zhang, and David Zhang. A trilateral weighted sparse coding scheme for real-world image denoising. In *Proceedings of the European Conference on Computer Vision*, pages 20–36, 2018.
- [YA02] Yongjian Yu and Scott T Acton. Speckle reducing anisotropic diffusion. *IEEE Transactions on Image Processing*, 11(11):1260–1270, 2002.
- [Yel83] John I Yellott. Spectral consequences of photoreceptor sampling in the rhesus retina. *Science*, 221(4608):382–385, 1983.
- [YFH20] Jiahui Yu, Yuchen Fan, and Thomas Huang. Wide activation for efficient image and video super-resolution. In *British Machine Vision Conference*, 2020.
- [YKY13] Ju Hong Yoon, Du Yong Kim, and Kuk-Jin Yoon. Gaussian mixture importance sampling function for unscented SMC-PHD filter. *Signal Processing*, 93(9):2664–2670, 2013.
- [YLL12] Miin-Shen Yang, Chien-Yo Lai, and Chih-Ying Lin. A robust EM clustering algorithm for Gaussian mixture models. *Pattern Recognition*, 45(11):3950–3961, 2012.
- [YQX⁺12] Fan Yang, Wenjian Qin, Yaoqin Xie, Tiexiang Wen, and Jia Gu. A shape-optimized framework for kidney segmentation in ultrasound images using NLTV denoising and DRLSE. *Biomedical Engineering Online*, 11(1):1–13, 2012.
- [YSM11] Guoshen Yu, Guillermo Sapiro, and Stéphane Mallat. Solving inverse problems with piecewise linear estimators: from Gaussian mixture models to structured sparsity. *IEEE Transactions on Image Processing*, 21(5):2481–2499, 2011.

- [YSM12] G. Yu, G. Sapiro, and S. Mallat. Solving inverse problems with piecewise linear estimators: from Gaussian mixture models to structured sparsity. *IEEE Transactions on Image Processing*, 21(5):2481–2499, 2012.
- [YTN⁺16] Kanefumi Yamashita, Shinsuke Takeno, Satoshi Nimura, Yoshikazu Sugiyama, Takayuki Sueta, Kenji Maki, Yoshiyuki Kayashima, Hironari Shiwaku, Daisuke Kato, Tatsuya Hashimoto, Takamitsu Sasaki, and Yuichi Yamashita. Gastric metastasis from salivary duct carcinoma mimicking primary gastric cancer. *International Journal of Surgery Case Reports*, 23:36–39, 2016.
- [YTN⁺18] Zhen Yu, Ee-Leng Tan, Dong Ni, Jing Qin, Siping Chen, Shengli Li, Baiying Lei, and Tianfu Wang. A deep convolutional neural network-based framework for automatic fetal facial standard plane recognition. *Journal of Biomedical and Health Informatics*, 22(3):874–885, 2018.
- [YWHM10] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, 19(11):2861–2873, 2010.
- [YWNL14] Hong-Ying Yang, Xiang-Yang Wang, Pan-Pan Niu, and Yang-Cheng Liu. Image denoising using nonsubsampling shearlet transform and twin support vector machines. *Neural Networks*, 57:152–165, 2014.
- [YY18] Yeo Hun Yoon and Jong Chul Ye. Deep learning for accelerated ultrasound imaging. In *International Conference on Acoustics, Speech and Signal Processing*, pages 6673–6676. IEEE, 2018.
- [YZX12] Chengpu Yu, Cishen Zhang, and Lihua Xie. An envelope signal based deconvolution algorithm for ultrasound imaging. *Signal processing*, 92(3):793–800, 2012.
- [ZBKT15] Ningning Zhao, Adrian Basarab, Denis Kouame, and Jean-Yves Tourneret. Joint Bayesian deconvolution and pointspread function estimation for ultrasound imaging. In *International Symposium on Biomedical Imaging*, pages 235–238. IEEE, 2015.
- [ZBLN97] Ciyou Zhu, Richard H Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software*, 23(4):550–560, 1997.
- [ZD18] Mingli Zhang and Christian Desrosiers. Structure preserving image denoising based on low-rank reconstruction and gradient histograms. *Computer Vision and Image Understanding*, 171:48–60, 2018.

- [ZFB⁺18] Yunfeng Zhang, Qinglan Fan, Fangxun Bao, Yifang Liu, and Caiming Zhang. Single-image super-resolution based on rational fractal interpolation. *IEEE Transactions on Image Processing*, 27(8):3782–3797, 2018.
- [ZGW⁺13] Zichun Zhong, Xiaohu Guo, Wenping Wang, Bruno Lévy, Feng Sun, Yang Liu, and Weihua Mao. Particle-based anisotropic surface meshing. *ACM Transactions on Graphics*, 32(4):99–1, 2013.
- [ZH16] Zichun Zhong and Jing Hua. Kernel-based adaptive sampling for image reconstruction and meshing. *Computer-Aided Geometric Design*, 43:68–81, 2016.
- [Zha18] Zijun Zhang. Improved Adam optimizer for deep neural networks. In *26th International Symposium on Quality of Service*, pages 1–2. IEEE, 2018.
- [ZLLQ19] Wenzhao Zhao, Qiegen Liu, Yisong Lv, and Binjie Qin. Texture variation adaptive image denoising with nonlocal PCA. *IEEE Transactions on Image Processing*, 28(11):5537–5551, 2019.
- [ZLRQ19] Zhemin Zhuang, Naihai Lei, Alex Noel Joseph Raj, and Shunmin Qiu. Application of fractal theory and fuzzy enhancement in ultrasound image segmentation. *Medical & Biological Engineering & Computing*, 57(3):623–632, 2019.
- [ZPIE17] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2223–2232, 2017.
- [ZWB⁺16] Ningning Zhao, Qi Wei, Adrian Basarab, Denis Kouamé, and Jean-Yves Tournet. Single image super-resolution of medical ultrasound images using a fast algorithm. In *International Symposium on Biomedical Imaging*, pages 473–476. IEEE, 2016.
- [ZZC⁺17] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.
- [ZZH19] Sikai Zhong, Zichun Zhong, and Jing Hua. Surface reconstruction by parallel and unified particle-based resampling from point clouds. *Computer-Aided Geometric Design*, 71:43–62, 2019.