



# Kernel PCA Based Non-Local Means Method for Speckle Reduction in Medical Ultrasound Images

Musab Elkheir Salih<sup>1</sup>, Xuming Zhang<sup>2\*</sup>, Mingyue Ding<sup>2</sup>

<sup>1</sup>Biomedical Department, College of Engineering, Sudan University of Science and Technology, Khartoum, Sudan

<sup>2</sup>School of Life Science and Technology, Huazhong University of Science and Technology, Wuhan, China

Email: 2yabaabobaba13@gmail.com, \*xmboshi.zhang@gmail.com, myding@mail.hust.edu.cn

**How to cite this paper:** Salih, M.E., Zhang, X.M. and Ding, M.Y. (2022) Kernel PCA Based Non-Local Means Method for Speckle Reduction in Medical Ultrasound Images. *Open Access Library Journal*, 9: e8618. <https://doi.org/10.4236/oalib.1108618>

**Received:** March 17, 2022

**Accepted:** April 21, 2022

**Published:** April 24, 2022

Copyright © 2022 by author(s) and Open Access Library Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## Abstract

The speckle noise is considered one of the main causes of degradation in ultrasound image quality. Many despeckling filters have been proposed, which are always making a trade-off between noise suppression and loss of information. A class of despeckling methods based Non-Local Means (NLM) algorithm is known to efficiently preserve the edges and all fine details of an image while reducing the noise. The core idea of NLM filter is to estimate the denoised pixel by performing a weighted average of similar patches in the neighborhood around the noisy pixel. However, the presence of noise degrades the similarity measurement process of the NLM and thereby decreases its efficiency. In this work, a novel despeckling scheme for ultrasound images is proposed, by introducing the kernel principal component analysis (PCA) to the NLM and computing the similarity in a high dimension kernel PCA subspace. The kernel representation is robust to the presence of noise and it can give better performance even under high noisy conditions. And it takes into account higher-order statistics of the pixels which can lead to accurate edge preservation. In this work, a novel despeckling scheme for ultrasound images is proposed using the kernel PCA-NLM extended to speckle noise model. The visual inspection and image metrics will show that the proposed filter is very competitive with respect to one of state-of-the-art methods, the Optimized Bayesian Non Local Means filter (OBNLN), in terms of low contrast object detectability, speckle noise suppression, edge's preservation.

## Subject Areas

Bioengineering, Biotechnology

## Keywords

US, Speckle Noise, NLM Kernel, OBNLN, PCA

## 1. Introduction

The US has long been recognized as a powerful tool used in the diagnosis and evaluation of different clinical examinations. Speckle is most often considered a dominant source of noise in US imaging [1]. In medical ultrasound imaging, the speckle noise masks small differences in grey level and spoils its visual observation quality and diagnostic procedure. Also, the presence of speckle noise limits the accuracy of computer-assisted methods such as feature extraction, analysis, recognition, and quantitative measurements problematic and unreliable [2]. Ultrasound image despeckling is a very challenging task in US images, therefore considerable effort has been spent over the last few decades in developing techniques to reduce the noise [3]. The speckle reduction aims to remove speckle noise as well as enhance image features and the detectability of small and low-contrast objects.

One of the state-of-the-art denoising methods is the NLM algorithm proposed by Buades [4]. The core idea of NLM filter to restore the noisy image is based on performing a weighted average of the most similar pixels to the pixel to be denoised are given the largest weights incorporating the non-local information [5]. Practically, the similarity is computed between equally sized neighborhoods around the pixels in consideration. These patches are more capable of capturing the image structures (e.g., texture) [6]. However, in the NLM filter as well as many denoising techniques, the suppression of the noise starts to degrade when the noise level increase. The presence of noise generally increases the dissimilarity between image patches and hence degrades the matching performance. This eventually leads to decreases in the efficiency of the NLM filter.

This work proposed modifications to the NLM method to handle the high speckle noise level in medical ultrasound imaging. The solution to this challenge is to design a robust and efficient NLM filter robust to presence of large noise. The question now becomes: how to determine the similarity weights in an optimal sense? Our key element is to design a high-performance NLM filter by improving the similarity computation between the image pixels, which is the core of the NLM algorithm.

To address this problem, one of the approaches to deal with noisy data is to learn the representation of the free noise and the most informative structures of the data matrix. Luckily, one of the promising techniques of machine learning is the kernel PCA, which can provide a solution to this problem. The kernel PCA can come up with a robust representation of the data in the high dimensional space. This work concerns the kernel PCA technique and can provide the ability to find the best image patch feature containing small noise, then calculate the similarities between patches around the pixels in that subspace. In this new space, the hidden information of the image patches due to the presence of the noise can be obvious. In sense of NLM, representing the image patch more concisely can help with their matching. These theoretical advantages of the kernel method motivated us to calculate similarity measurement in high dimensional space to

improve the efficiency of the NLM in the presence of large noise. This will be a super useful method for medical ultrasound images, due to the narrow dynamic range of the gray levels and the typically high level of noise which is introduced at all stages of image acquisition, especially the speckle noise [7]. To our knowledge, no research exists addressing the kernel PCA-NLM framework to derive a NLM filter for ultrasound despeckling.

The initial formulation of the NLM filter relies on the assumption of an additive white Gaussian noise model (AWGN). However, the noise which degrades ultrasound images is signal-dependent and more complex. The distribution of noise in ultrasound images has been largely studied in the literature and many models have been proposed. A more general speckle noise model was first introduced for ultrasound image denoising by Loupas *et al.* [8]. This model has been considered in many studies. Accordingly, given this model, in this paper we introduce a novel restoration scheme for US images, using the kernel PCA motivation for the NLM filter. The remainder of this paper is structured as follows. Section 2 covers the basic idea of the NLM in more detail, in addition to briefly reviewing different approaches to optimize the parameters of the NLM to improve its quality or speed in the author's own terms. Section 3 gives a brief overview of the main speckle reducing categories, providing a short description of the principle behind each family. Section 4 introduces the common speckle noise models. The theory underlying the OB-NLM is presented in details in Section 5, which adapted to speckle noise and archived best despeckling results among the others.

Section 6 explains the idea behind the learning representation theory. Section 7 describes the implementation of the kernel PCA-NLM speckle reduction method. In addition to applications of the proposed de-speckle information about artificial phantom data and real ultrasound images and definitions of image quality metrics are presented in details. Section 8 shows the experimental promising results of the proposed method. The theoretical interpretations of the findings are due to the superiority of the proposed discussed in Section 9. Finally, a conclusion and further works are needed to bring an improvement to performance of our speckle filter in Section 10.

Appendix A contains the important martial, that we do not put in the body of the paper which is necessary to understand the data analysis. Appendix A contains all the mathematical background of linear PCA, and procedure to compute the PCA. Lastly, the kernel PCA method which has been developed as a non-linear version of the PCA to extract feature in higher dimensional space is explained.

## 2. The Non Local Means Definition and Parameters

Given a discrete noisy image  $v = \{v(i) | i \in \Omega\}$  the NLM scans the huge space of the image to enhance the redundancy, instead of update the pixel value as an average of its limited local four or eight neighborhood as local image filter. The

contribution of the all others pixels is weighted according to their similarities. Given these weights the estimated value  $NL(v)(i)$  is computed as a weighted average of all pixels in the image [9]. The noise goes down with square root of the number of times of averaging [10].

$$NL(v)(i) = \sum_{j \in \Omega} w(i, j)v(j) \quad (1)$$

NLM looks for similar pixels around and average them for denoising the current pixel. To find the similar pixels, the Euclidean distance ( $d(i, j)$ ) between pixels at location  $i, j$  is computed. The  $d(i, j)$  does not depend only on the pixels  $v(i)$ , and  $v(j)$  but also on its surrounding windows  $x_i$  and  $x_j$ , respectively. By using patch-wise representation the pixel is no longer is a single feature, but a more informative high dimensional data point. This window size should be large enough to be robust to the noise and at the same time to be able to preserve fine details. Indeed, the patch size reflects the scale of the “noise” compared to the image resolution [11].

Based on theoretical grounding nonlocal philosophy, the search window ( $V$ ) of radius ( $t$ ) should be large enough to be robust to noise and at the same time to be able to take care of the details and fine structure [11]. In [12], the authors have noticed that, choosing locally the best search windows can control the filtering results. Smaller search windows are not robust enough to noise and in case of strongly textured images [13]. As search window grows, there are slightly differences among the root mean square error (RMSE), but sharply increasing in the operation period. Thus, restricting the search of patches to a window is the common practice that, besides the speed-up, the result is visually better [5]. With a smaller search window size, the small features will be preserved [14].

In the original NLM the Euclidean distance using a Gaussian weighted function, is given by:

$$d(i, j) = \int_{\Omega} G_a(\tau) |v(i + \tau) - v(j + \tau)|^2 d\tau \quad (2)$$

Normally the dissimilarity of the patches is not a binary thing, and the dissimilarity of the patches is transformed into weighting function, that argues which patches are considered as a similar or a dissimilar. To achieve this weighting a Gaussian weighting function is used, which is defined as:

$$w(i, j) = \frac{1}{Z(i)} \exp\left(-\frac{d(i, j)}{h^2}\right) \quad (3)$$

where the parameter  $h$  controls the decay of the weighting function and  $Z(i)$  is a normalizing term defined by:

$$Z(i) = \sum_j \exp\left(-\frac{d(i, j)}{h^2}\right) \quad (4)$$

In the natural images there are some points which are more distinct and unique than the others, and do not appear anywhere. On the other hand there patches that have more close patches than others. So the normalizing term is

needed to count and guarantee the following conditions:

$$\begin{aligned} 0 &\leq w(i, j) \leq 1 \\ \sum_j w(i, j) &= 1 \end{aligned} \quad (5)$$

The exponential weighting function can be considered as a decreasing function depending on the similarity of the patches. If a particular local difference has a large magnitude, then the value of the  $w(i, j)$  will be small and therefore, that measurement will have little effect on the output image. The NLM uses exponential weighting function of the Euclidean distance between their arguments, which still assigns positive weights to dissimilar neighbourhoods. When weights are very small, the estimated pixel intensities can be severely biased due to many small contributions [15].

The parameter  $h$ , namely the width of the exponential function, quantifies how fast the weights decay with increasing dissimilarity of respective patches. The larger the parameter the smoother the output, and the edges will be blurred, and on the other hand, choosing a very small  $h$  leads to noisy results identical to the input [11]. The  $h$  parameter is typically optimized manually in the NLM algorithm [16]. The simplest and most common one is to set a single  $h$  for the whole image. The  $h$  parameter is an important parameter, and best value of  $h$  is roughly proportional to noise standard deviation [4].

### 3. A View on Despeckle Filtering Methods

Several different despeckle filtering methods are used based upon different mathematical models of the phenomenon [17]. Speckle is considered to be a deterministic process (not random) because when an object is imaged under the same operating conditions no changes in the speckle pattern occur. For these reasons, speckle cannot be reduced by signal averaging over time. Unlike the additive white Gaussian noise model adopted in most denoising methods, ultrasound imaging requires specific filters due to the signal dependent nature of the speckle intensity. In what follows, we give a brief overview on the main speckle reducing methods, providing a short description of the principle behind each family and some of its limitations are discussed. Filters that are used widely in both SAR and ultrasound imaging include the following categories: adaptive filters, homomorphic, anisotropic diffusion, and wavelet filtering [18]. A number of adaptive speckle filters have been proposed, and they are widely used in US image restoration because they are easy to implement and control [19]. The most cited and applied filters in this category include the Lee [20] [21] [22], Frost [23], Kuan [24], and Gamma maximum a posteriori (MAP) filters [25]. Many improvements of these classical filters have been proposed since [26]. The Lee and Frost filters have the same structure, whereas the Kuan filter is a generalization of the Lee filter. Both filters form the output image by computing the central pixel intensity inside a filter-moving window, which is calculated from the average intensity values of the pixels and a coefficient of variation inside the moving window [1]. In comparison with non-adaptive speckle filters (The best-known

non-adaptive filters are those based on the use of the mean or the median), adaptive speckle filters are more successful in preserving subtle image information. Adaptive filters use weights that are dependent on the degree of speckle in the image, whereas non-adaptive filters use the same set of weights over the entire image [19]. A homomorphic filtering is used in ultrasound to sharpen features and flatten speckle variations in an image [1]. This form of filtering performs image despeckling by computing the fast Fourier transform (FFT) of the logarithmic compressed image, applying a denoising homomorphic filter function, and then performing the inverse FFT of the image. The homomorphic filter function may be constructed either using a band-pass Butterworth or a high-boost Butterworth filter [27]. Speckle reduction filtering in the wavelet domain is based on the idea of the Daubechies and Symlet wavelet and on soft-thresholding denoising [1]. Wavelet filtering exploits the decomposition of the image into the wavelet basis where only the useful wavelet coefficients are utilized and zeroes out the other wavelet coefficients to despeckle the image. Different wavelet thresholding approaches can be used [28]. Diffusion filters filter perform contrast enhancement and remove noise from an image by modifying the image via solving a PDE. The diffusion coefficient in these filters serves as the edge detector, producing high values at features and low values in homogeneous regions. So in the region where the edge is high, the diffusion will be suppressed and vice versa. Speckle reducing anisotropic diffusion (SRAD) is formulated as an efficient anisotropic diffusion despeckling technique. The SRAD not only preserves edges but also enhances edges by inhibiting diffusion across edges and allowing diffusion on either side of the edge. SRAD is adaptive and does not utilize hard thresholds to alter performance in homogeneous regions or in regions near edges and small features. This technique was compared with the Frost filter Lee filter and homomorphic filtering and documented that anisotropic diffusion performed better [29]. Unlike the adaptive speckle filters, all the considered PDE-based approaches produce a family of resulting images based on an iterative diffusion process. However, it has no rational criteria to select the optimal stop criteria. So it may has a limitation in retaining subtle features such as small cysts and lesions in ultrasound images Nevertheless, meaningful structural details are unfortunately removed during a large number of iterations [11] [19]. The previously mentioned approaches for speckle reduction are based on the so-called locally adaptive recovery paradigm [30]. Compared with them, the NLM algorithm relies on a L2-norm between two image patches instead of pixel comparison and the pattern redundancy is not restricted to be local. This strategy leads to competitive results when compared to most of the state-of-the-art methods. However, the noise of ultrasound images cannot be considered as an AWGN. To address this problem, considering the Bayesian formulation [31] and the Loupas noise model [8] the OBNLM algorithm for speckle reduction in ultrasound images was proposed by Coupé *et al.* [32]. This formulation improve the denoising performances of the NLM filter for the speckle noise removing while preserving meaningful edges in compared to the compared the original

NLM filter, adaptive filter, and SRAD filter where no stop criterion is needed [32] [33]. Among the many existing despeckle methods to tackle image despeckle, we may quote OBNLM technique, which has provided satisfactory results among them.

## 4. Noise Models

The distribution of ultrasonic speckle noise has been largely studied for many years. A realistic modeling of noise statistics of ultrasound images cannot be easily exhibited, considering the complex image formation process. Speckle is described as one of the more complex image noise models; unlike thermal and readout noise, it is non-Gaussian, object dependent, with its variance being proportional to the local field intensity [34]. In the literature there are many models have been proposed, what follows is a briefly review of the main three models for the amplitude distribution of the backscattered ultrasound.

### 4.1. The Additive Noise Model

Noise reduction filters work under the assumption that the only degradation present in an image is additive Gaussian noise ( $\eta_a$ ) [35] [36] then

$$v = u + \eta_a \quad (6)$$

where:

$v$  is the recorded noisy image is the sum of the clean image  $u$  and  $\eta_a$  is the AWGN.

Consider a discrete description of the, in term of the NLM, the expectation ( $E$ ) of the Gaussian weighted Euclidean distance of the intensity grayscale vectors  $v(x_i)$  and  $v(x_j)$  can be written as:

$$E \left\| v(x_i) - v(x_j) \right\|_{2,a}^2 \quad (7)$$

The Euclidean distance measure is quite adapted to an additive white noise, which alters the distance between image patches, so the above equation can be written as:

$$E \left\| u(x_i) - u(x_j) \right\|_{2,a}^2 + 2\sigma_{\eta_a}^2 \quad (8)$$

where  $v(x_i)$  and  $v(x_j)$  are respectively, the original and noisy neighbourhoods and  $\sigma_{\eta_a}^2$  is the additive noise variance. Here, twice the amount of noise is presented assuming that the two neighbourhoods receive the same amount of noise. The Gaussian noise assumption is widely prevalent in the context of other noise model. Many methods dealing with Poisson noise rely on variance stabilization techniques as in [37] [38]. By using the Anscombe transform and treat the processed image as if it was corrupted by a Gaussian noise [39].

### 4.2. Multiplicative Speckle Noise Model

Clearly, the signal-dependent nature of the speckle must be taken into account to design an efficient speckle reduction filter. Jain *et al.* [40] [41] [42] have ex-

plained that the speckle noise model for US signal at the output of the receiver demodulation module of the US imaging system may be approximated as multiplicative as

$$v = u\eta_m + \eta_a \quad (9)$$

where:

$\eta_m$  is the multiplicative noise.

Since the effect of additive noise (such as sensor noise) is considerably small compared to that of multiplicative noise (coherent interfering), the (9) can be approximated by

$$v = u\eta_m \quad (10)$$

Due to the limited dynamic range of commercial display monitors, ultrasound imaging systems compress the large echo signal to fit in the display range [1] [2]. Using the mathematical logarithmic transformation, the multiplicative speckle noise model in (10) can be converted into an additive Gaussian noise [7].

$$\log(v) = \log(u) + \log(\eta_m) \quad (11)$$

The (11) can also be written as

$$f^l = g^l + \eta_a^l \quad (12)$$

where:

$v^l, u^l, \eta_a^l$  are the observed noisy image, clean image, and noise component after the logarithmic transformation, respectively.

There exists a class of approaches for additive noise reduction that use a multiplicative model of speckled image formation and take advantage that the logarithm compression of ultrasound images transforms the speckle into an additive Gaussian noise, such as the homomorphic despeckling methods in the wavelet denoising domain [27] [43] [44] [45] [46]. Furthermore, the assumption that the reconstructed positron emission tomography (PET) images are corrupted by the Gaussian noise is widely prevalent [47]. Many methods dealing with Poisson noise use the Anscombe transform to treat the processed image as if it was corrupted by AWGN noise [39].

### 4.3. Rayleigh Distribution Model

Wagner *et al.* [48] [49] [50] [51] have showed that the histogram of amplitudes within the resolution cells of the envelope-detected radio frequency (RF) signal backscattered from a uniform area with a sufficiently high scatterer density has a Rayleigh distribution with mean proportional to the standard deviation.

$$P_{RL}(v(i, j)) = \frac{v(i, j)}{\psi} \exp\left(\frac{-v(i, j)^2}{2\psi}\right) \quad (13)$$

where  $v(i, j)$  is a pixel intensity at the  $i, j$ -th position, and  $\psi$  is the shape parameter of  $P_{RL}$  related to the mean square scattering amplitude of the tissue in the scattering medium [48].



The logarithmical transformation may also modify the characteristics of the Rayleigh speckle noise model. As a result, the speckle noise becomes very close to white Gaussian noise corresponding the uncompressed Rayleigh [2] [42] [52].

#### 4.4. General Signal-Dependent Noise Model

Loupas *et al.* [8] suggested that the linear relationship between the mean and the standard deviation valid for Rayleigh distributed speckle no longer holds for ultrasound images. However, the linear relation between the mean and the variance ensures that speckle specifications of these images fit the signal-dependent noise model of the form:

$$v = u + \sqrt{v}\eta_a \quad (14)$$

Loupas *et al.* have shown that their model offered a better fitting to data than the multiplicative model or the Rayleigh model and contrary to the white Gaussian noise model. The Loupas noise model is image-dependent. The following subsections details the OBNLM algorithm which utilized this speckle noise model.

### 5. Optimized Bayesian NL-Means Filter (OBNLM)

In this section, the OBNLM is presented. In [31] [53], a Bayesian formulation of the NLM filter is used to derive a new speckle filter. The Kervrann [31] generalized NLM filter is called Bayesian NLM mean (BNL) filter. In [14], an algorithm based on maximum likelihood estimation (MLE) which allows one to deal with noises other than Gaussian, e.g., speckle noise was introduced. The Bayesian NLM framework is based on the probabilistic intensity similarity measure, in contrast to the conventional NLM relies on  $L_2$  norm distance metrics which have a fixed structure [31] [53]. This metrics is formulated as the likelihood that the noise distributions of the two intensity observations are the same and correspond to the same scene radiance value. It considers how intensities fluctuate in the imaging system by embedding the effect of noise. This approach considered the effect the image noise distributions as a source of useful information, rather than attempting to remove the noise. Such that the similarity is high when the two intensities are both well within the noise distributions of certain true intensities, and become significantly lower otherwise. The optimal Bayesian estimator for the  $v(i)$  can be written as

$$\hat{u}_{opt} v(i) = \frac{\sum_{j=1}^m v(j) p(v(i)|v(j)) p(v(j))}{\sum_{j=1}^m p(v(i)|v(j)) p(v(j))} \quad (15)$$

Using the Bayes's and marginalization rules, and  $p(v(i)|v(j))$  and  $p(v(i))$  respectively denote the distribution of  $v(i)/v(j)$  and prior distribution of  $v(i)$ .

Compared to the classical NLM formulation, the OBNLM introduced the blockwise implementation and a new statistical distance for patch comparison (Pearson distance) for weight computation [32]. The restoration of a block  $v(B)$  based on a Bayesian NLM scheme is given by

$$NL(v)(B_i) = \frac{\frac{1}{|V_i|} \sum_{j=1}^{|V_i|} p(v(B_i) | v(B_j)) p(v(B_j)) v(B_j)}{\frac{1}{|V_i|} \sum_{j=1}^{|V_i|} p(v(B_i) | v(B_j)) p(v(B_j))} \quad (16)$$

where:

$V_i$  is the search window centered on the pixel  $i$ .

$B_i$  is the block centered on the pixel  $i$ .

$v(B_i)$  is the vector containing the intensities of the block  $B_i$ .

$NL(v)(B_i)$  is the vector containing the restored value of  $B_i$ .

$p(v(B_i)|u(B_i))$  and  $p(v(B_i))$  respectively denote the distribution of  $v(B_i)$  conditionally to  $u(B_i)$  and the prior distribution of  $v(B_i)$ .

In the case of a white Gaussian noise, the likelihood  $p(u(B_i)|u(B_i))$  is proportional to

$$p(v(B_i) | v(B_j)) \propto \exp\left(-\frac{\|v(B_i) - v(B_j)\|^2}{h^2}\right) \quad (17)$$

The likelihood can be factorized for a block considers the noise model (7) is obtained by:

$$p(v(B_i) | v(B_j)) \propto \exp\left(-\frac{\|v(B_i) - v(B_j)\|^2}{v(B_j) h^2}\right) \quad (18)$$

So instead of the usual  $L_2$ -norm, the Pearson distance defined as

$$d_p(v(B_i) - v(B_j))^2 = \frac{(v(B_i) - v(B_j))^2}{v(B_j)} \quad (19)$$

This similarity measure is better for US images than the  $L_2$ -norm because it takes into account the impact of the nature of the speckle noise in an ultrasound image.

## 6. The Representation Learning Theory

The performance of machine learning methods is heavily influenced by the different forms of data representation on which they are applied [54] [55]. There are many applications of representation learning algorithms such as speech recognition, object recognition, visualization, data compression and natural language processing. We refer the reader to [54] for extensive details about these learning methods. PCA is the most widely used technique for dimension reduction. it is an unsupervised linear method used to learn a lower representation of the data by a linear projection of the input data features onto a set of new features which are a linear function of all of the original features [56] [57] [58]. PCA is useful for data visualization tasks, by projecting a very large dimensional data (for example gene expression) onto a lower dimensional space that can be more readily and quickly visualize how these data are related to each other and get better sense of what data looks like [59] [60]. The PCA transform can be

used for image compression without losing important properties of the data [61] [62]. Lower dimensional framework is used in face recognition to transform the image into what is called Eigen-faces. Using fewer Eigen images is computational efficiency and provides better ability to learn and recognize new faces [63]. The image (e.g., face photo), can be represented as a linear combination of basis images. The number of basis images can be much smaller than the original collection. By using few PC's, we can capture more than 50% of the variance in the image. This presentation is very useful in face recognition algorithm [63].

In 1965, G. Golub and W. Kahan introduced the singular value decomposition (SVD) as a decomposition technique for calculating the singular values, pseudo-inverse and rank of a matrix [64]. PCA can be efficiently computed via the SVD of the data matrix. In case of having large matrix, the SVD is an efficient, numerically stable and fast technique to solve the eigenvalue problem by matrix factorization [65]. Frequently use low-rank approximation of the data matrix is considered as noise reduction process. By discarding the later PCs which are treated as noise factors and projecting back the dataset into the original observation space will remove the noise. The latent structure of the data might be masked by noisy dimensions. In image denoising, it may be an efficient way to transform the local image patches in different representation coefficients. The idea behind this strategy is to decompose the local image patches, select the clean coefficients, and then reconstruct them. Several authors have shown that to decompose an image in wavelet base is superior for image denoising [66] [67] [68] [69] [70]. The independent components (ICs) was applied on the on local image patches to derive locally basis set [71]. The free noise structures of input data can be obtained by preserving the most informative features and removing the outliers which are associated with the least important dimensions [72]. The PCA can be used to find the less noise pre-image of the image patch in the (Eigen) space, and then calculate similarity measurement of the NLM filter in lower-dimensional space. It has shown in [73], that PCA combined with patches was an efficient filter for images corrupted by Poisson noise and outperformed the Poisson-NLM method. Based on PCA-NLM, two authors [16] [74] have proposed simultaneously similar idea for Gaussian noise reduction. They suggested firstly projecting the noisy patch to the most important PC's first and then computing the similarity. Their hypothesis has significantly improved the denoising effect and the computation cost. Furthermore, computing distances in few dimensions and eliminating fraction of patches around the pixel under processing using the  $L_2$  of the rank-1 approximations has accelerated the NLM algorithm [16]. However, the PCA is not always good enough for learning representation of the data. The PCA is purely second order representation and processing of the data, but it is observed that more information in the images is in the higher order of the data. In other words, the Eigen space only utilizes the gray scale information variance that may weakly related to the properties of the natural data. However, presence of noise and distortion affects the calculated principal components and hence the overall performance of SVD [75] [76]. Our

work [77] [78] have showed, that the SVD dimension reduction is powerful only when images are contaminated by small amounts of noise. However, when the noise level is particularly high, the treatment relying on linear PCA transform is no longer relevant. The kernel method is one of the promising methods, which have attracted significant attention in the area of the machine learning. It is like opposite way to the dimension reduction approach, that instead of small number of PCs the kernel method can get benefit from the more features. It comes up with a different idea for similarity measurement in high dimensional space, helps in keeping the similar thing together and dissimilar thing apart. Appendix A details the theory of the kernel method. It is first appeared in the form of support vector machine (SVM) which is one of the powerful binary classification algorithms. For non-linearly separable data, the kernel method helps researchers to build an efficient SVM linear classifier in high-dimensional feature space [79]. The data points become linearly separable in the higher dimensional space, which is not the case in the lower-dimensional space. In the original space, the higher-dimensional data points are not visualized, but they are projected down to the lower dimensional space. The kernel technique is not restricted only to SVM. However various algorithms in machine learning can be enhanced with the use of the kernel method. In face recognition, the kernel technique is usually used to transform the image into what is called Eigen-faces in order to enhance the matching process and improving the ability to learn to recognize new faces [80]. Kernel PCA is a nonlinear version for computing singular vectors in high dimensional space [81]. That, the kernel PCA can come with a better encoding of the information. Furthermore it is more robust to the presence of the noise than the linear PCA [77] [82] [83] [84]. So, the kernel method representation can produce a robust subspace to the presence of the noise as well a rich informative encoding of the image patches. As it will be shown in the following sections, this approach will highlight similarities between different patches, which are the core of the NLM algorithm. Therefore, increase the performance of NLM algorithm.

## 7. Material and Methods

This section presents a detailed description of the implementation of the proposed speckle noise reduction scheme. The core of the proposed method concentrates on finding the right representation of the patches which will highlight the matching process of the noisy image patches. Thus, better denoising performance can be obtained when similarity between pixels is computed in free noise space. The nonlinear projective approach produced by kernel PCA can remove the difficulties faces the NLM in the presence of large noise by using an appropriate kernel function and represent image data to a relatively high dimensional space. As it will be shown in the result section, this approach removes the speckle noise and enhanced the contrast of the input US images. The information about the US images used in the experiment, and the quality image metrics is presented below. Our main methodology involves the comparison of results produced by the OBNLM and kernel PCA-NLM.

## 7.1. Details Kernel PCA-NLM Method

In this framework, we focus on obtaining the most compact patch representations in a higher-dimensional manifold. Then calculating the similarity of the image patches in that space, which consequently contribute to effective NLM denoising effect. The detailed kernel PCA-NLM can be described as follows:

- All the NLM parameters ( $f$ ,  $t$ ,  $h$ ), in addition to the width of Gaussian kernel ( $h_k$ ) are optimized empirically to achieve the best despeckling results.
- Stack all the pixels of each patch in a single row vector  $x_i \in \mathbb{R}^{1 \times n}$ , where  $n = (2f + 1) \times (2f + 1)$  centered on pixel  $i$ . And then construct the database matrix  $X \in \mathbb{R}^{m \times n}$  from the number ( $m$ ) of the patches vectors in the search window. Where numbers of columns in  $X$  are the numbers of pixels in the image patch and numbers of rows in  $X$  are the numbers all patches in the search window.
- Construct the kernel matrix  $K \in \mathbb{R}^{m \times m}$  by computing the distance between all the pairs of features in the  $X$  matrix using the RBF which the most popular kernel function, and closes to our work for what is the similarity could be, since it decays with  $L_2$  norm [85]. The size of the  $K_{ij}$  scales with the square of the number of patches in the search window [86]. So the kernel SVD can extract a larger number of principal components that can exceed the data dimensionality [87].
- Compute the kernel SVD and get the direction of the singular vectors that learn the higher-dimensional space
- Perform SVD on the  $K_{ij}$  matrix using (A-1) to find an adaptive basis in the projection data set. See Appendix A which describes the SVD in more details.
- Project  $K$  into that high dimensional space using (19) to get the coordinates of rows of  $K_{ij}$  in the space of singular vectors.

$$X^z V^z = U^z \Sigma^z \quad (20)$$

- A more general speckle noise model was first introduced for ultrasound image denoising by Loupas *et al.* [8]. This model and has been considered in many studies. This model is particularly suitable for our purpose, since the presented despeckle filter works on the images as displayed by the US machine, rather than the envelope detected echo signal. In particular, this model employed to implement the relevant solution scheme, which reflects the nature of noise distribution in the ultrasound images. At this stage, given the speckle noise model, it now becomes possible to devise the kernel PCA motivation for NLM filter for speckle reduction in US imaging. Calculating the weights of image patches based on the Euclidean distance in case of the noise is correlated decreases the NLM denoising performance as it was proved in [22]. Clearly, the similarity weights should be adapted to the image in order to achieve maximal improvement. Accordingly, at this stage, we apply the NLM method and calculate distances in high dimensional space. By taking the speckle noise statistics into account a robust similarity computation can be obtained. We follow exactly the same NLM routine except that the Euclidean metric in (7) is replaced by

$$\left\| \frac{-1}{v(j)h^2} v(x^z(i)) - v(x^z(j)) \right\|^2 \quad (21)$$

• where:  $v(x^z(i))$  is the projections of  $x_i$  onto the higher-dimensional space ( $z$ ) using the kernel PCA. Where:  $v(x^z(i))$  is the projections of  $v(x_i)$  onto the  $d$ -dimensional space using the SVD. The weights  $v(x^z(i))$  are just the corresponding row in  $U$  multiplied by diagonal elements of  $\Sigma$ , see the Appendix A. The  $v(x^z(i))$  representation is insensitive noise and preserve the patch structure as well. Some of the patches in the  $X$  matrix are similar to  $v(x^z(i))$ , patch in the center of the neighborhood, but a number of different patches as well. The closest match of other patches to the  $v(x^z(i))$  in the neighborhood will have high weight.

What follows is information about the US images used in the experiment, and definitions of quality metrics to quantifying the performance of the proposed despeckle and the OBNLM schemes.

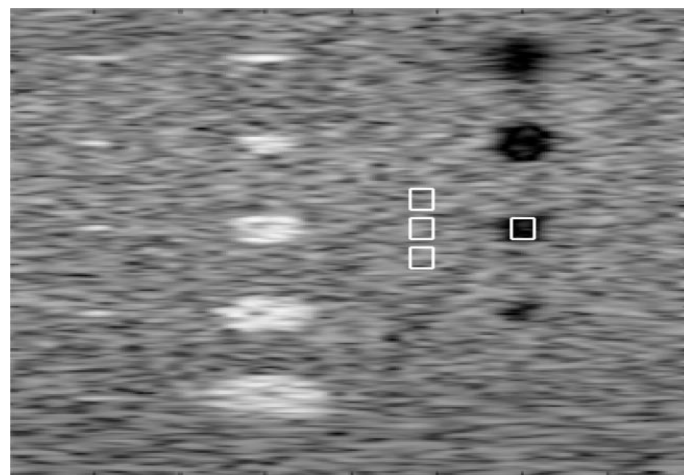
## 7.2. Details of the OBNLM Method

The parameters values of the OBNLM (such as search area size, patch size and smoothing parameter) were set as [32].

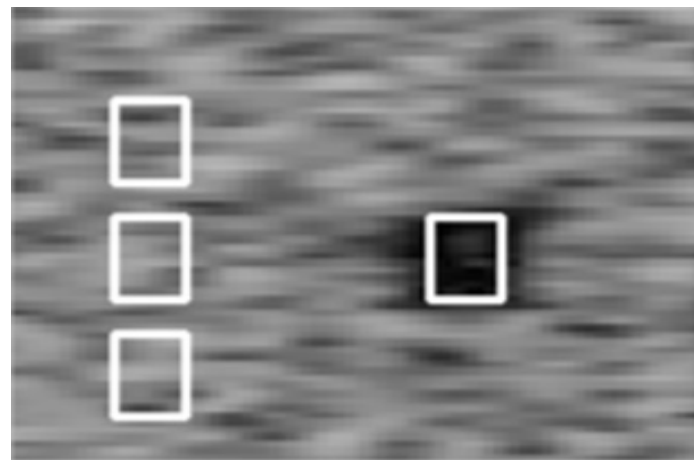
## 7.3. Data Set for Testing the Proposed De-Speckle Method and the SRAD

### 7.3.1. Synthetic US Image

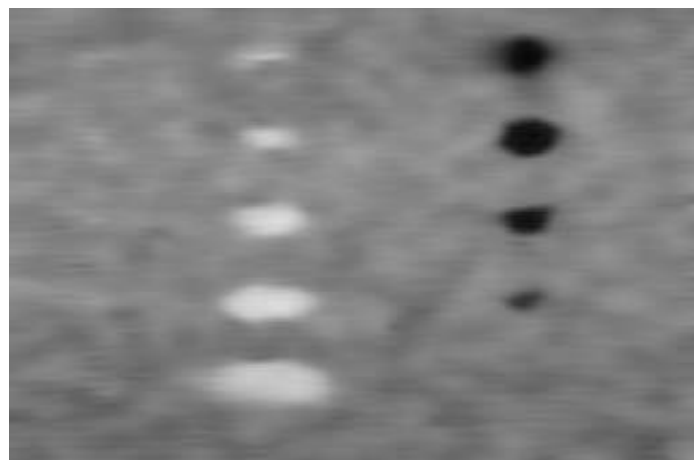
A cyst phantom is used in the experiments as it is shown in **Figure 1(a)**. The synthetic ultrasound image is simulated by using FIELD II software [88]. The cyst contains five high scattering target points, five echoic regions (has white color on ultrasound) of six, five, four, three, two mm diameter, and five anechoic (*i.e.*, it absorbs all the sound, and it appears black) of six, five, four, three, two mm diameter water-filled cysts. The size of the original phantom image is large ( $672 \times 504$ ); instead, a smaller size ( $256 \times 256$ ) is used to speed up the denoising process.



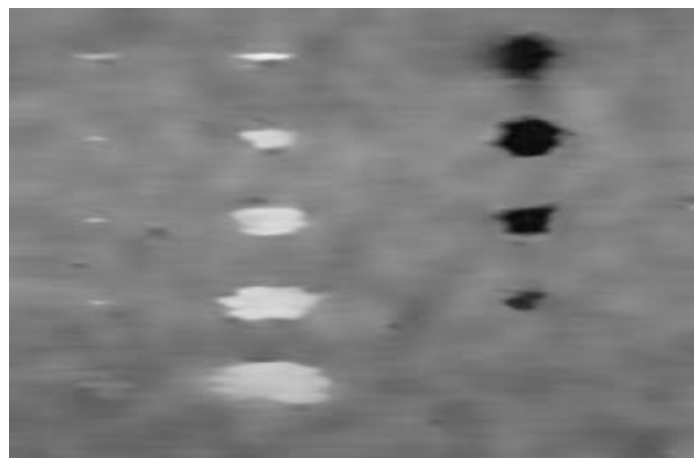
(a)



(b)



(c)



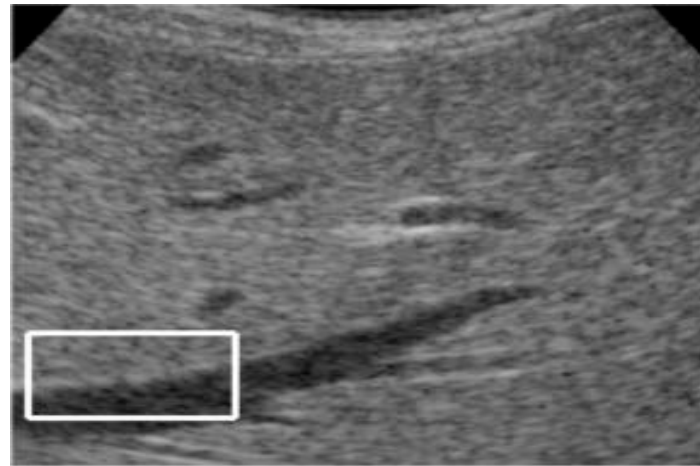
(d)

**Figure 1.** Despeckling results for Cyst Phantom: (a) The simulated cyst phantom. (b) The zoomed rectangular white boxes indicate ROIs used to calculate ENL and CNR. (c) De-speckled image Using SRAD; (d) De-speckled image using kernel PCA-NLM.

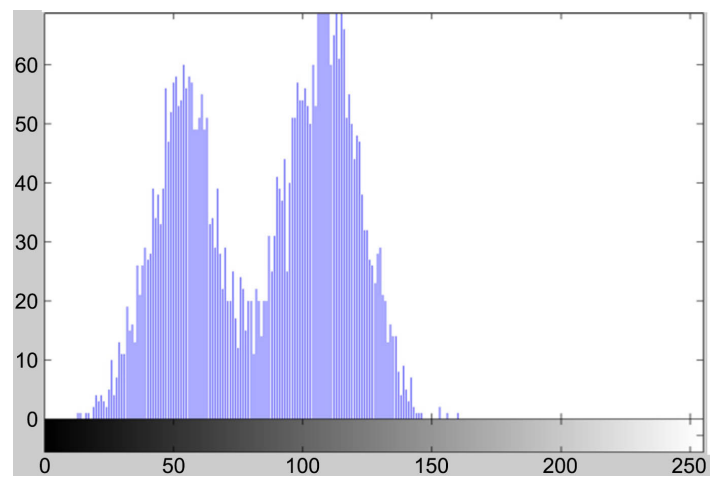
### 7.3.2. Real Ultrasound Images

Four arbitrary real US images (US001, US002, US003, and US004) are used in

the experiments (Figures 2-5). The sizes of the images are:  $287 \times 259$ ,  $290 \times 233$ ,  $381 \times 301$ , and  $455 \times 345$ , respectively. The US002 is carotid artery cross-sectional US image; see Figure 3(a). The US001, US003, US004 are liver US images, as they is shown in Figure 2(a), Figure 4(a) and Figure 5(a).



(a)

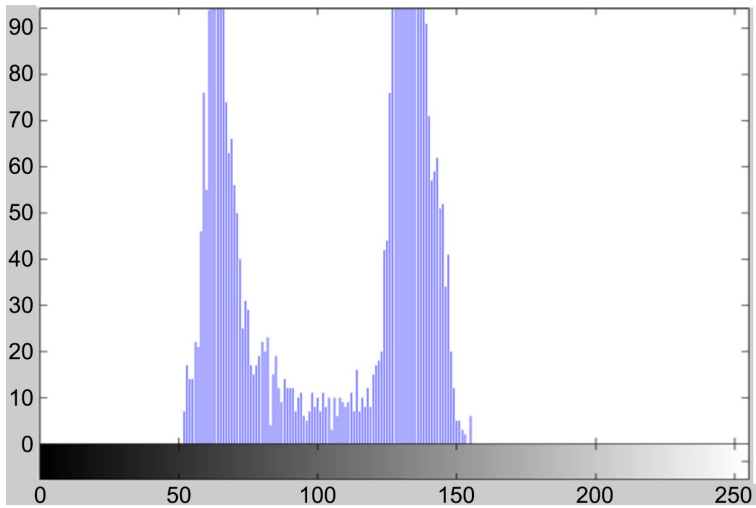


(b)



(c)

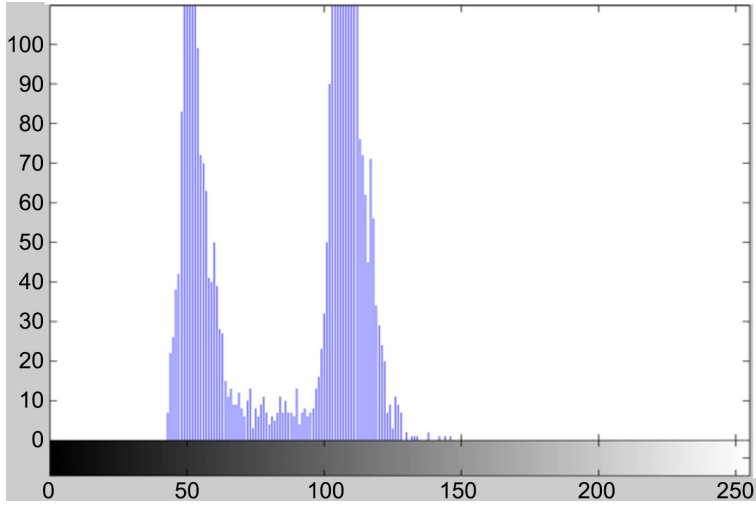




(d)



(e)

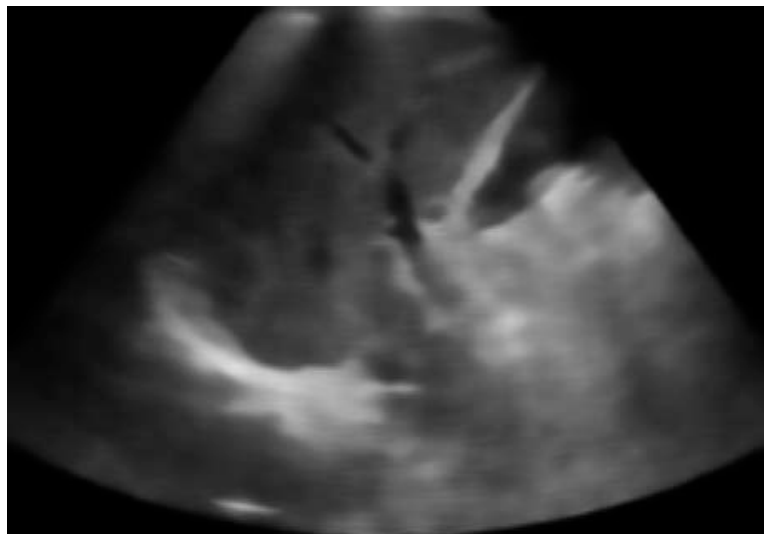


(f)

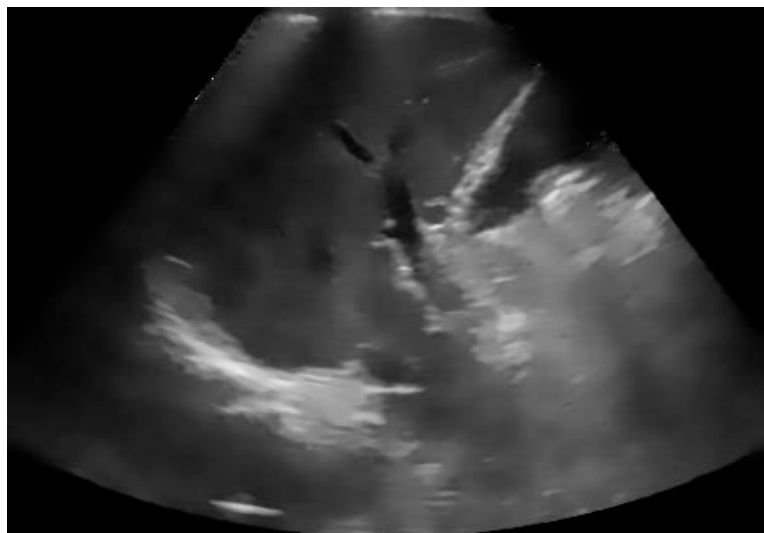
**Figure 2.** The original US001 image and the results of application of the two de-speckle filters are given in (a)-(c). The corresponding HOs are given in (d)-(f).



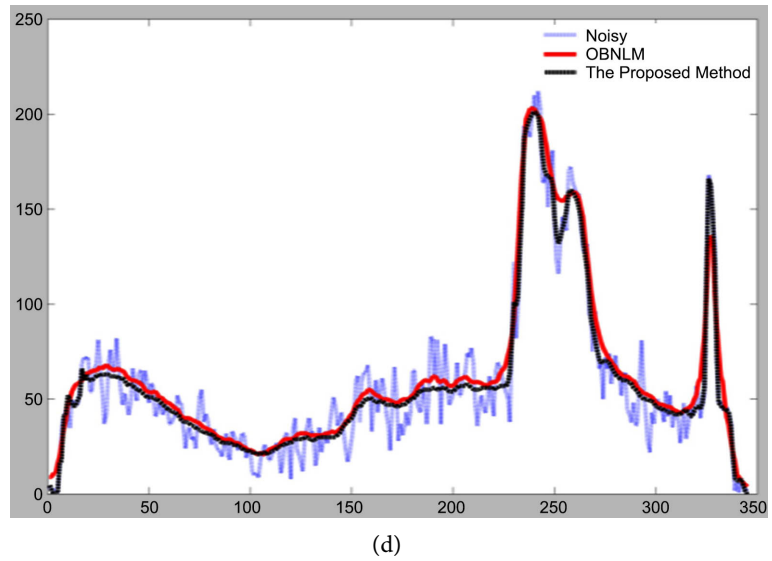
(a)



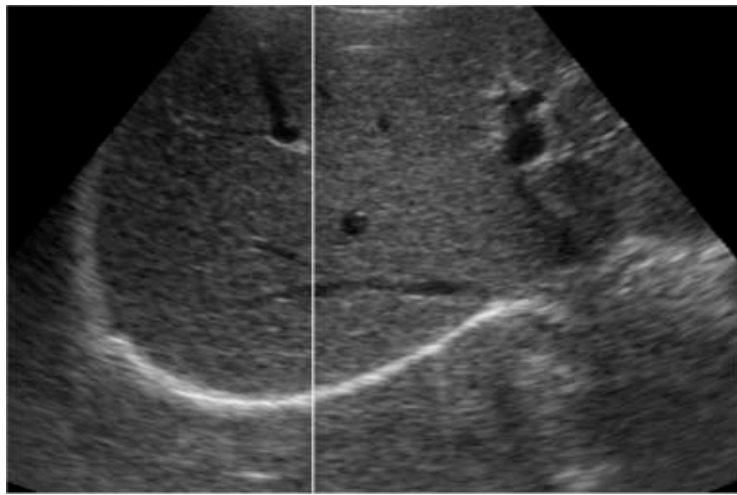
(b)



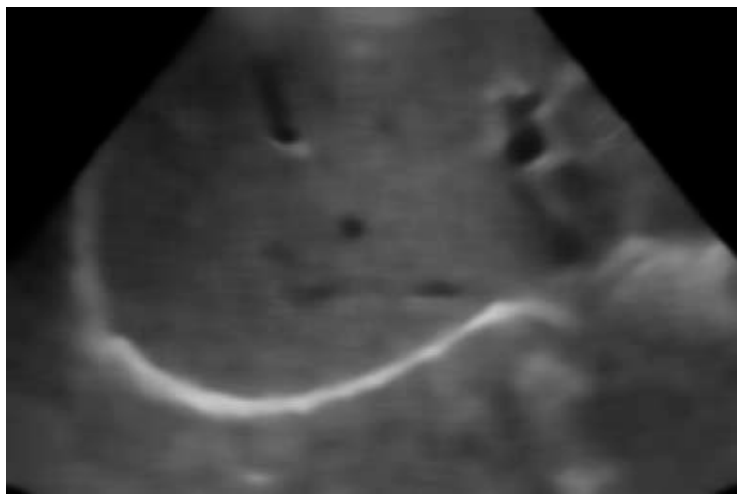
(c)



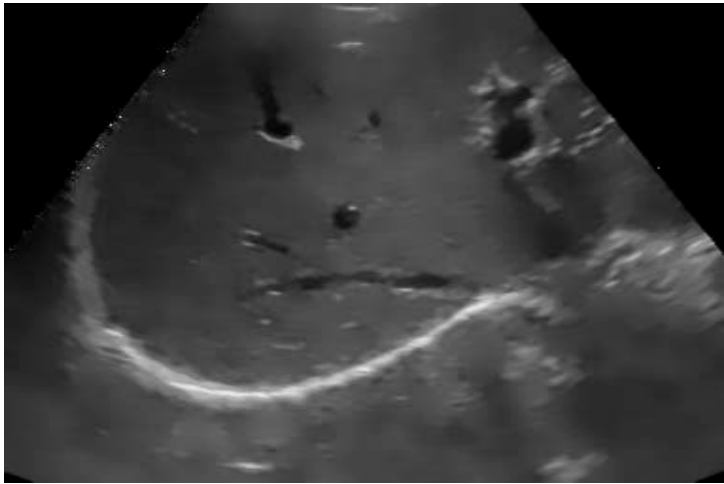
**Figure 3.** The original US002 image and the results of application of the two de-speckle filters are given in (a)-(c). The corresponding LPs are given in (d).



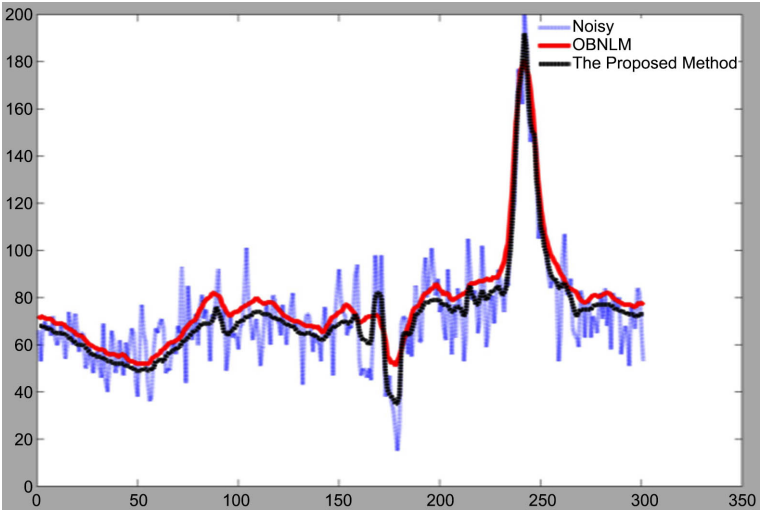
(a)



(b)

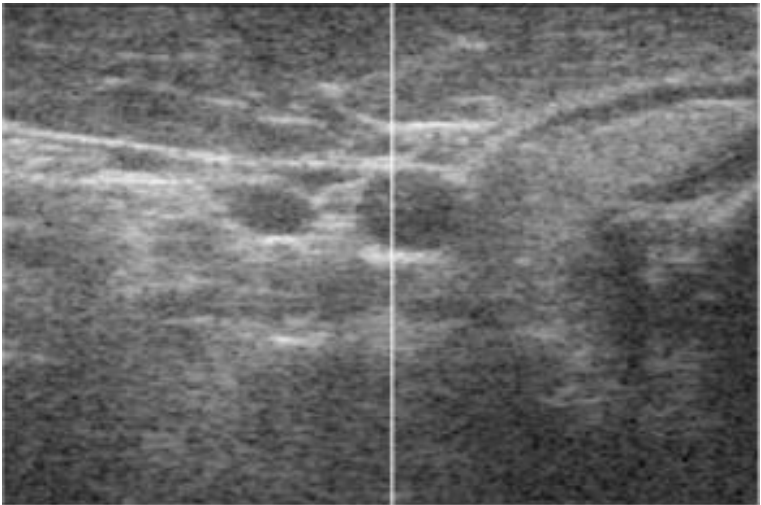


(c)



(d)

**Figure 4.** The original US003 image and the results of application of the two de-speckle filters are given in (a)-(c). The corresponding LPs are given in (d).



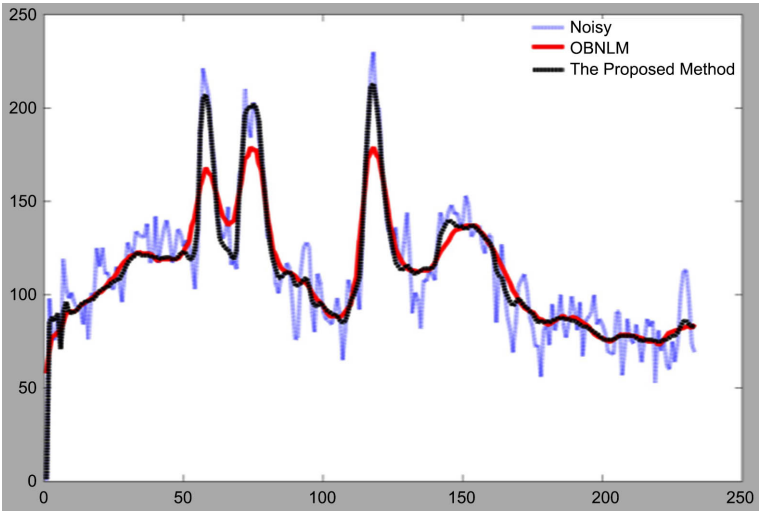
(a)



(b)



(c)



(d)

**Figure 5.** The original US004 image and the results of application of the two de-speckle filters are given in (a)-(c). The corresponding LPs are given in (d).

## 7.4. Image Quality Metrics

Usually speckle reduction methods require a trade-off between noise reduction and edge preservation. All the speckle reduction methods are known as edge preserving [29]. Since of the edges are important information used to make the objects in the ultrasound image accessible for the diagnosis and evaluation purposes. In ultrasound imaging the noise-free reference image is not known, a comparison of images before and after processing is often the best way to evaluate performance of a filtering algorithm. Here, for the quantitative and the visual comparisons, three criteria, including the ENL, the CNR, and the visual inspection are used as no-reference metric. Each criterion reflects one aspect of the despeckling purpose, and definition of these measures is defined in detail below.

### 7.4.1. Equivalent Number of Looks

The smoothness of a homogeneous region of an ultrasound image can be evaluated by ENL, which it is computed as:

$$\text{ENL} = \left( \frac{\mu_H^2}{\sigma_H^2} \right) \quad (22)$$

where  $\mu_H^2$  and  $\sigma_H^2$  are the mean and variance of the uniform region [89]. The higher ENL value is corresponding to get lowest  $\sigma_H^2$  and to preserve  $\mu_H^2$ . In other words, ENL usually increases with noise reduction.

### 7.4.2. Contrast-to-Noise-Ratio

The CNR or lesion signal-to-noise-ratio is a quantitative measure of the contrast between an image object (for example, lesion or cyst) and an area of background speckle noise [89] [90], and it is defined by:

$$\text{CNR} = \frac{|\mu_o - \mu_s|}{\sqrt{\sigma_o^2 + \sigma_s^2}} \quad (23)$$

where:  $\mu_o$  and  $\sigma_o^2$  are the mean and variance of intensities of pixels of the object;  $\mu_s$  and  $\sigma_s^2$  are the mean and variance of intensities of pixels in a speckle noise surrounding the image object. A larger CNR correlated with the better contrast.

### 7.4.3. Visual Evaluation

The visual evaluation is defined as the ability of an expert to extract useful anatomical features from an ultrasound image, and it is subjective to the observer's variability [1]. The evaluation is carried out in terms of a histogram overlap (HO) and line profile (LP) of selected pixels. The image histogram shows the gray-scale intensity values distribution of pixels of the ultrasound image. Ideally, for superior contrast, there is no overlap present between the histogram two regions of different gray scale distributions [91]. The LP is used as a tool to compare the ability of the de-speckle filters to smooth the noise and show the details

of the small or large clearly observable regions in the ultrasound images. The region that shows distinct boundaries (large details) is often looked as being sharp. On the other hand, the presence of blur produces images of low contrast [92].

## 8. Experimental Results

In this section, we evaluate the performance of our speckle reduction and the OB-NLM filters on the data set, and the reasons for these findings are presented. All the experiments and visualization of the results are implemented using MATLAB software (Math works).

### 8.1. Cyst Phantom Results

Here, we propose to compare different filters with experiments on synthetic data. In addition to the visual observation, the CNR and ENL are chosen for objective comparisons. The despeckled cyst images of the two filters are shown in **Figures 1(c)-(d)**. The visual inspection shows that, the proposed filter remove most speckle in the homogenous region and highlight important features of the image, and even the small objects like the target points of the cyst phantom. Such details are lost (cannot be visualized) in the despeckled image using the OB-NLM, see **Figure 1(c)**.

To calculate the ENL, and the CNR the third anechoic cyst from top and its three surrounding normal tissue areas in the background are selected as the region of interests (ROIs), which are indicted by the white boxes as it is shown in **Figure 1(b)**. The ENL values are averaged over the three homogeneous ROIs on the background. **Table 1** shows the ENL and CNR values obtained for each method. The ENL quantifies the effectiveness of the speckle suppression performance. The results demonstrate that the two methods remove the speckle well and greatly increased original ENL of original images. But the kernel PCA-NLM still holds high ENL values, surpassing the OB-NLM method. The CNR is a metric of contrast preservation performances. The CNR increases are obtained after filtering using kernel PCA-NLM and OB-NLM, which indicates the filtered images have more contrast with the original one. But the proposed filter has the higher CNR value. We can conclude that our method offers better output in terms of ENL and CNR. These results collaborate with the visual observation mentioned before.

**Table 1.** The Performance of the two de-speckling filters on the cyst phantom in terms of ENL, and CNR.

Cyst	ENL	CNR
	14.4916	6.3188
OB-NL	32.5983	11.4030
Kernel PCA-NLM	34.6456	13.6482

## 8.2. Real Ultrasound Images

**Figure 2(a)** shows the HO of the two distinct regions (enclosed by a white box) of the US001 image. **Figure 2(c)** and **Figure 2(e)** show the visual denoising results of the OBNLM and the kernel PCA-NLM filters on the US001 image, respectively. It is clearly both despeckling methods reduce the histogram overlapping and hence improve the image contrast as it is shown in **Figure 2(d)** and **Figure 2(f)**. However, the kernel PCA-NLM enhances the overall contrast more. **Figures 3(b)-5(b)** show the denoising results of the OBNLM filter on the US002, US003 and US004 images, respectively. The despeckle output of these images, using the kernel PCA-NLM method, are shown in **Figures 3(c)-5(c)**. It is obviously both despeckling methods reduce the speckle noise and improve the image contrast. In the OBNLM output, only the sharper boundaries are preserved, while the weak edges are blurred, that better visual result of the US images details are as assessed by our method. On the other hand, much better visualization of the US images details is obtained with the proposed method. It is obviously. From the top to bottom through the images, a white vertical line indicates the position where the intensity profile is taken. The LPs of the original, the despeckled ultrasound images applying the OBNLM and the proposed filter are plotted using black dashed, and blue and red solid line styles, respectively. The kernel PCA-NLM strategy achieved competitive results. The LPs of the US002, US003 and US004 US images are given in **Figures 3(d)-5(d)**, respectively. The LPs show that the two filters produce smoother gray-value line profiles and preserve the boundaries. But the OBNLM profile is a lesser sharp, in some region along the white line, in compare to the corresponding profile using the proposed method. In other words, in some areas smallest details are visualized for LPs of the OBNLM. In contrast, the LPs associated with the kernel PCA-NLM show much greater details.

## 9. Discussions

The above experimental results have proved that the two filters were an efficient despeckle way, while it is not surprising, the kernel PCA-NLM essentially achieved the best despeckling effect. It was able to reduce noise in a homogenous region and enhanced the US the image contrast. The kernel PCA obtained better visual inspection as well as higher ENL and CNR values surpassing the OBNLM method which it still damaged image details, especially, the tiny boundary. In this section, the causes of the finding of all images used in experiments are discussed.

The OBNLM despeckle algorithm was considered as one of the state-of-the-art despeckling methods based on Bayesian estimator framework to derive the NLM method. The OBNLM filter is based on clever idea which takes the noise properties into account to calculate the similarity between the image patches using the Bayesian frame. The probabilistic similarity measure implemented in the in the OBNLM filter better indicate the confidence of a match than the  $L_2$  norm, in a meaningful way, that leads to improve performance of NLM filter. Furthermore,



it employed a new statistical distance measure which considers the impact of the noise as a relevant criterion for patch comparison. The main drawback of the NLM filter is that it determines the pixel similarity based on the noisy image patches thereby leading to inaccurate filtering. This drawback existed in the OBNLM filter which made a strong assumption that a noisy observation itself provides a good approximation of the true intensity signal [93]. Since the weights are computed directly based on the observed noisy image patches, which can deviate much from the true value, as a result, the weights become sensitive to the noise and lessens the efficiency of the OBNLM.

It is not surprising the kernel PCA-NLM achieved the best despeckling performance, as we have mentioned previously, that minimizing the  $L_2$  cost improves NLM algorithm performance. The similarity computation operation in the NLM filters is enhanced with the more cleanly and informative representation of the image patch in the higher dimensional space. The kernel SVD can extract a larger number of principal components that can exceed the data dimensionality [86] [87]. In terms of NLM, the high-dimensional feature induced by kernel PCA, the number of dimensions is equal to the number of patches in the neighbourhood, which can be quite large than the original patch size. This higher dimensional representation helps in keeping the similar things together and dissimilar things apart in the nonlinear subspaces produced by the kernel PCA. This is similar to what we mention in section 6 that the kernel method enhances the classification performance of the SVM algorithm by creating the kernel based SVM classifier [79].

The higher representation produced by the kernel PCA is robustness to the presence of noise. When adding more PCs using the kernel PCA, more reliable features are reconstructed without getting the noise yet. Since there are many structures choosing from many PCs [94].

The higher dimensional encoding provided by the kernel concept gives the potential of getting clean features and removing some of the eigenvectors where the noisy part of the data resides [83].

Since the kernel space produces more features sitting on their correct positions and provides less noisy features. This higher-dimensional space provided by the kernel PCA is similar to what is called super resolution idea [95], where the smear structures like edges or corners within the low resolution image become sharper in the high-resolution image. That, the kernel method scales up of the features dimension and leads to produce a high-resolution image from a blurred one [86]. This is another interpretation of why the higher dimensional feature space since leads to a better denoising. The kernel method is capable of capturing part of the higher-order statistics which are particularly important for encoding image structure [86] [96] [97]. The polynomial kernel is another popular kernel [98], and the high dimensional space by this kernel may contain all possible interactions of among the pixels of different orders as it is shown in (A-24). In other words, the kernel representation also can take into account higher-order

statistics of the pixels like that the relation between the more than two pixels in edges or curves [99]. All that we have mentioned before are our interpretations of why the kernel PCA-NLM have provided a better speckle noise reduction and enhanced identification of the edges in compare to the OBNLM method.

## 10. Conclusions and Future Works

In this paper, an extension of the NLM is proposed for US images degraded by speckle noise. Our novel restoration scheme for US images is guided by learning a good encode of the image patches using the kernel PCA-NLM adapted to the statistics of speckle noise. The similarity computation operation in the NLM filters is enhanced with the informative representation of the image using Kernel PCA. Experiments were carried out on phantom data and real US images. In comparison to the OBNLM technique, the experimental results have shown that the proposed filter brought significant improvement in terms of ENL, and CNR and visual inspection.

Although the proposed method performs better than the OBNLM techniques, the scope for improvement still exists. What follows are some suggestions to improve the quality of this work. Capturing the higher-order statistics for the image processing by the kernel method is more efficient and accurate but has computational complexity. Since kernel PCA requires storing and manipulating the kernel matrix the size of which is the square of the number of training patches in the search window. So in future work, modifications may be incorporated to reduce the computation time. To make the algorithm faster methods such as the Kernel Hebbian Algorithm (KHA) can be used [86] [100]. Enriching our understanding of higher-order statistics, using the kernel method, can help in learning a better expressive image representation. The same study can be also conducted by investigating different kernel representations which can give a better representation of the pattern of the images, and hence improve the performance of the proposed denoising method.

## Conflicts of Interest

The authors declare no conflicts of interest.

## References

- [1] Loizou, C.P. and Pattichis, C.S. (2008) Despeckle Filtering Algorithms and Software for Ultrasound Imaging. Synthesis Lectures on Algorithms and Software in Engineering, Vol. 1, Morgan & Claypool, San Rafael, 1-166.  
<https://doi.org/10.2200/S00116ED1V01Y200805ASE001>
- [2] Abd-Elmoniem, K.Z., Youssef, A. and Kadah, Y.M. (2002) Real-Time Speckle Reduction and Coherence Enhancement in Ultrasound Imaging via Nonlinear Anisotropic Diffusion. *IEEE Transactions on Biomedical Engineering*, **49**, 997-1014.  
<https://doi.org/10.1109/TBME.2002.1028423>
- [3] Sanchez, J.R. and Oelze, M. (2009) An Ultrasonic Imaging Speckle-Suppression and Contrast-Enhancement Technique by Means of Frequency Compounding and Coded

- Excitation. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, **56**, 1327-1339. <https://doi.org/10.1109/TUFFC.2009.1189>
- [4] Buades, A., Coll, B. and Morel, J.-M. (2005) A Review of Image Denoising Algorithms, with a New One. *Multiscale Modeling & Simulation*, **4**, 490-530. <https://doi.org/10.1137/040616024>
- [5] Duval, V., Aujol, J.-F. and Gousseau, Y. (2011) A Bias-Variance Approach for the Nonlocal Means. *SIAM Journal on Imaging Sciences*, **4**, 760-788. <https://doi.org/10.1137/100790902>
- [6] Darbon, J., Cunha, A., Chan, T.F., Osher, S. and Jensen, G.J. (2008) Fast Nonlocal Filtering Applied to Electron Cryomicroscopy. *5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, Paris, 14-17 May 2008, 1331-1334. <https://doi.org/10.1109/ISBI.2008.4541250>
- [7] Boukerroui, D., Noble, J.A. and Brady, M. (2003) Velocity Estimation in Ultrasound Images: A Block Matching Approach. In: *Information Processing in Medical Imaging*, Springer, Berlin, 586-598. [https://doi.org/10.1007/978-3-540-45087-0\\_49](https://doi.org/10.1007/978-3-540-45087-0_49)
- [8] Loupas, T., McDicken, W. and Allan, P. (1989) An Adaptive Weighted Median Filter for Speckle Suppression in Medical Ultrasonic Images. *IEEE Transactions on Circuits and Systems*, **36**, 129-135. <https://doi.org/10.1109/31.16577>
- [9] Buades, A., Coll, B. and Morel, J.-M. (2010) Image Denoising Methods. A New Nonlocal Principle. *SIAM Review*, **52**, 113-147. <https://doi.org/10.1137/090773908>
- [10] Gonzalez, R.C. and Richard, E. (2002) *Woods, Digital Image Processing*. Prentice Hall Press, Hoboken.
- [11] Guo, Y., Wang, Y. and Hou, T. (2011) Speckle Filtering of Ultrasonic Images Using a Modified Non Local-Based Algorithm. *Biomedical Signal Processing and Control*, **6**, 129-138. <https://doi.org/10.1016/j.bspc.2010.10.004>
- [12] Kervrann, C. and Boulanger, J. (2006) Optimal Spatial Adaptation for Patch-Based Image Denoising. *IEEE Transactions on Image Processing*, **15**, 2866-2878. <https://doi.org/10.1109/TIP.2006.877529>
- [13] Brox, T., Kleinschmidt, O. and Cremers, D. (2008) Efficient Nonlocal Means for Denoising of Textural Patterns. *IEEE Transactions on Image Processing*, **17**, 1083-1092. <https://doi.org/10.1109/TIP.2008.924281>
- [14] Deledalle, C.-A., Denis, L. and Tupin, F. (2009) Iterative Weighted Maximum Likelihood Denoising with Probabilistic Patch-Based Weights. *IEEE Transactions on Image Processing*, **18**, 2661-2672. <https://doi.org/10.1109/TIP.2009.2029593>
- [15] Goossens, B., Luong, Q., Pizurica, A. and Philips, W. (2008) An Improved Non-Local Denoising Algorithm. *2008 International Workshop on Local and Non-Local Approximation in Image Processing (LNLA 2008)*, Lausanne, 23-24 August 2008, 143-156.
- [16] Orchard, J., Ebrahimi, M. and Wong, A. (2008) Efficient Nonlocal-Means Denoising Using the SVD. *15th IEEE International Conference on Image Processing*, San Diego, 12-15 October 2008, 1732-1735. <https://doi.org/10.1109/ICIP.2008.4712109>
- [17] Kanevsky, M.B. (2008) *Radar Imaging of the Ocean Waves*. Elsevier, Amsterdam.
- [18] Loizou, C.P., Pattichis, C.S., Christodoulou, C.I., Istepanian, R.S., Pantziaris, M. and Nicolaides, A. (2005) Comparative Evaluation of Despeckle Filtering in Ultrasound Imaging of the Carotid Artery. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, **52**, 1653-1669. <https://doi.org/10.1109/TUFFC.2005.1561621>
- [19] Mather, P. and Tso, B. (2010) *Classification Methods for Remotely Sensed Data*. CRC Press, Boca Raton.

- [20] Lee, J.-S. (1980) Digital Image Enhancement and Noise Filtering by Use of Local Statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2**, 165-168. <https://doi.org/10.1109/TPAMI.1980.4766994>
- [21] Lee, J.-S. (1981) Refined Filtering of Image Noise Using Local Statistics. *Computer Graphics and Image Processing*, **15**, 380-389. [https://doi.org/10.1016/S0146-664X\(81\)80018-4](https://doi.org/10.1016/S0146-664X(81)80018-4)
- [22] Lee, J.-S. (1986) Speckle Suppression and Analysis for Synthetic Aperture Radar Images. *Optical Engineering*, **25**, Article ID: 255636. <https://doi.org/10.1117/12.7973877>
- [23] Frost, V.S., Stiles, J.A., Shanmugan, K.S. and Holtzman, J.C. (1982) A Model for Radar Images and Its Application to Adaptive Digital Filtering of Multiplicative Noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-4**, 157-166. <https://doi.org/10.1109/TPAMI.1982.4767223>
- [24] Kuan, D.T., Sawchuk, A., Strand, T.C. and Chavel, P. (1987) Adaptive Restoration of Images with Speckle. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **35**, 373-383. <https://doi.org/10.1109/TASSP.1987.1165131>
- [25] Lopes, A., Touzi, R. and Nezry, E. (1990) Adaptive Speckle Filters and Scene Heterogeneity. *IEEE Transactions on Geoscience and Remote Sensing*, **28**, 992-1000. <https://doi.org/10.1109/36.62623>
- [26] Lopes, A., Nezry, E., Touzi, R. and Laur, H. (1993) Structure Detection and Statistical Adaptive Speckle Filtering in SAR Images. *International Journal of Remote Sensing*, **14**, 1735-1758. <https://doi.org/10.1080/01431169308953999>
- [27] Saniie, J., Wang, T. and Bilgutay, N.M. (1989) Analysis of Homomorphic Processing for Ultrasonic Grain Signal Characterization. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, **36**, 365-375. <https://doi.org/10.1109/58.19177>
- [28] Suri, J.S. (2008) *Advances in Diagnostic and Therapeutic Ultrasound Imaging*. Artech House, Norwood.
- [29] Yu, Y. and Acton, S.T. (2002) Speckle Reducing Anisotropic Diffusion. *IEEE Transactions on Image Processing*, **11**, 1260-1270. <https://doi.org/10.1109/TIP.2002.804276>
- [30] Elad, M. (2002) On the Origin of the Bilateral Filter and Ways to Improve It. *IEEE Transactions on Image Processing*, **11**, 1141-1151. <https://doi.org/10.1109/TIP.2002.801126>
- [31] Kervrann, C., Boulanger, J. and Coupé, P. (2007) Bayesian Non-Local Means Filter, Image Redundancy and Adaptive Dictionaries for Noise Removal. In: *Scale Space and Variational Methods in Computer Vision*, Springer, Berlin, 520-532. [https://doi.org/10.1007/978-3-540-72823-8\\_45](https://doi.org/10.1007/978-3-540-72823-8_45)
- [32] Coupé, P., Hellier, P., Kervrann, C. and Barillot, C. (2009) Nonlocal Means-Based Speckle Filtering for Ultrasound Images. *IEEE Transactions on Image Processing*, **18**, 2221-2229. <https://doi.org/10.1109/TIP.2009.2024064>
- [33] De Fontes, F.P.X., Barroso, G.A., Coupé, P. and Hellier, P. (2011) Real Time Ultrasound Image Denoising. *Journal of Real-Time Image Processing*, **6**, 15-22. <https://doi.org/10.1007/s11554-010-0158-5>
- [34] Uzan, A., Rivenson, Y. and Stern, A. (2013) Speckle Denoising in Digital Holography by Nonlocal Means Filtering. *Applied Optics*, **52**, A195-A200. <https://doi.org/10.1364/AO.52.00A195>
- [35] Dougherty, G. (2009) *Digital Image Processing for Medical Applications*. Cam-

- bridge University Press, Cambridge. <https://doi.org/10.1017/CBO9780511609657>
- [36] Marques, O. (2011) Practical Image and Video Processing Using MATLAB. Wiley, Hoboken. <https://doi.org/10.1002/9781118093467>
- [37] Boulanger, J., Kervrann, C., Bouthemy, P., Elbau, P., Sibarita, J.-B. and Salamero, J. (2010) Patch-Based Nonlocal Functional for Denoising Fluorescence Microscopy Image Sequences. *IEEE Transactions on Medical Imaging*, **29**, 442-454. <https://doi.org/10.1109/TMI.2009.2033991>
- [38] Zhang, B., Fadili, J.M. and Starck, J.-L. (2008) Wavelets, Ridgelets, and Curvelets for Poisson Noise Removal. *IEEE Transactions on Image Processing*, **17**, 1093-1108. <https://doi.org/10.1109/TIP.2008.924386>
- [39] Anscombe, F.J. (1948) The Transformation of Poisson, Binomial and Negative-Binomial Data. *Biometrika*, **35**, 246-254. <https://doi.org/10.1093/biomet/35.3-4.246>
- [40] Dutt, V. (1995) Statistical Analysis of Ultrasound Echo Envelope. Biophysical Sciences—Biomedical Imaging—Mayo Graduate School.
- [41] Jain, A.K. (1989) Fundamentals of Digital Image Processing. Prentice-Hall, Inc., Hoboken.
- [42] Zong, X., Laine, A.F. and Geiser, E.A. (1998) Speckle Reduction and Contrast Enhancement of Echocardiograms via Multiscale Nonlinear Processing. *IEEE Transactions on Medical Imaging*, **17**, 532-540. <https://doi.org/10.1109/42.730398>
- [43] Odegard, J.E., Guo, H., Lang, M., Burrus, C.S., Wells Jr., R.O., Novak, L.M. and Hiatt, M. (1995) Wavelet-Based SAR Speckle Reduction and Image Compression. *SPIE's 1995 Symposium on OE/Aerospace Sensing and Dual Use Photonics*, Orlando, 17-21 April 1995, 259-271. <https://doi.org/10.1117/12.210843>
- [44] Gagnon, L. and Smaili, F.D. (1996) Speckle Noise Reduction of Airborne SAR Images with Symmetric Daubechies Wavelets. In: *Aerospace/Defense Sensing and Controls*, International Society for Optics and Photonics, Bellingham, 14-24. <https://doi.org/10.1117/12.241168>
- [45] Bovik, A.C. (2009) The Essential Guide to Image Processing. Academic Press, Cambridge.
- [46] Jain, A. (1989) Fundamental of Digital Image Processing. Prentice-Hall, Englewood Cliffs.
- [47] Zhou, Y., Endres, C.J., Brašić, J.R., Huang, S.-C. and Wong, D.F. (2003) Linear Regression with Spatial Constraint to Generate Parametric Images of Ligand-Receptor Dynamic PET Studies with a Simplified Reference Tissue Model. *Neuroimage*, **18**, 975-989. [https://doi.org/10.1016/S1053-8119\(03\)00017-X](https://doi.org/10.1016/S1053-8119(03)00017-X)
- [48] Wagner, R.F., Smith, S.W., Sandrik, J.M. and Lopez, H. (1983) Statistics of Speckle in Ultrasound B-Scans. *IEEE Transactions on Sonics and Ultrasonics*, **30**, 156-163. <https://doi.org/10.1109/T-SU.1983.31404>
- [49] Burckhardt, C.B. (1978) Speckle in Ultrasound B-Mode Scans. *IEEE Transactions on Sonics and Ultrasonics*, **25**, 1-6. <https://doi.org/10.1109/T-SU.1978.30978>
- [50] Chen, Y., Yin, R., Flynn, P. and Broschat, S. (2003) Aggressive Region Growing for Speckle Reduction in Ultrasound Images. *Pattern Recognition Letters*, **24**, 677-691. [https://doi.org/10.1016/S0167-8655\(02\)00174-5](https://doi.org/10.1016/S0167-8655(02)00174-5)
- [51] Balocco, S., Gatta, C., Pujol, O., Mauri, J. and Radeva, P. (2010) SRBF: Speckle Reducing Bilateral Filtering. *Ultrasound in Medicine & Biology*, **36**, 1353-1363. <https://doi.org/10.1016/j.ultrasmedbio.2010.05.007>
- [52] Chen, S., Yang, X., Yao, L.P. and Sun, K. (2005) Total Variation-Based Speckle Re-

- duction Using Multi-Grid Algorithm for Ultrasound Images. In: *Image Analysis and Processing-ICIAP 2005*, Springer, Berlin, 245-252. [https://doi.org/10.1007/11553595\\_30](https://doi.org/10.1007/11553595_30)
- [53] Zhong, H., Li, Y. and Jiao, L. (2009) Bayesian Nonlocal Means Filter for SAR Image Despeckling. *2nd IEEE Asian-Pacific Conference on Synthetic Aperture Radar*, Xian, 26-30 October 2009, 1096-1099. <https://doi.org/10.1109/APSAR.2009.5374145>
- [54] Bengio, Y., Courville, A. and Vincent, P. (2013) Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **35**, 1798-1828. <https://doi.org/10.1109/TPAMI.2013.50>
- [55] van der Meij, J. and de Jong, T. (2004) Learning with Multiple Representations. *Annual Meeting of the American Educational Research Association*, San Diego, 12-16 April 2004, 16.
- [56] Hotelling, H. (1933) Analysis of a Complex of Statistical Variables into Principal Components. *Journal of Educational Psychology*, **24**, 417. <https://doi.org/10.1037/h0071325>
- [57] Pearson, K. (1901) LIII. On Lines and Planes of Closest Fit to Systems of Points in Space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, **2**, 559-572. <https://doi.org/10.1080/14786440109462720>
- [58] Jolliffe, I. (2005) Principal Component Analysis. Wiley Online Library. <https://doi.org/10.1002/0470013192.bsa501>
- [59] Ivosev, G., Burton, L. and Bonner, R. (2008) Dimensionality Reduction and Visualization in Principal Component Analysis. *Analytical Chemistry*, **80**, 4933-4944. <https://doi.org/10.1021/ac800110w>
- [60] Wall, M.E., Rechtsteiner, A. and Rocha, L.M. (2003) Singular Value Decomposition and Principal Component Analysis. In: Berrar, D.P., Dubitzky, W. and Granzow, M., Eds., *A Practical Approach to Microarray Data Analysis*, Kluwer, Norwell, 91. [https://doi.org/10.1007/0-306-47815-3\\_5](https://doi.org/10.1007/0-306-47815-3_5)
- [61] Smith, L.I. (2002) A Tutorial on Principal Components Analysis. Cornell University, Ithaca, 51-52.
- [62] Du, Q. and Fowler, J.E. (2007) Hyperspectral Image Compression Using JPEG2000 and Principal Component Analysis. *IEEE Geoscience and Remote Sensing Letters*, **4**, 201-205. <https://doi.org/10.1109/LGRS.2006.888109>
- [63] Turk, M. and Pentland, A. (1991) Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, **3**, 71-86. <https://doi.org/10.1162/jocn.1991.3.1.71>
- [64] Golub, G. and Kahan, W. (1965) Calculating the Singular Values and Pseudo-Inverse of a Matrix. *Journal of the Society for Industrial & Applied Mathematics, Series B: Numerical Analysis*, **2**, 205-224. <https://doi.org/10.1137/0702016>
- [65] Golub, G.H. and Van Loan, C.F. (1996) Matrix Computations. Johns Hopkins University Press, Baltimore, 374-426.
- [66] Chang, S.G., Yu, B. and Vetterli, M. (2000) Spatially Adaptive Wavelet Thresholding with Context Modeling for Image Denoising. *IEEE Transactions on Image Processing*, **9**, 1522-1531. <https://doi.org/10.1109/83.862630>
- [67] Li, X. and Orchard, M.T. (2000) Spatially Adaptive Image Denoising under Overcomplete Expansion. *IEEE International Conference on Image Processing*, Vancouver, 10-13 September 2000, 300-303.
- [68] Coifman, R. and Donoho, D. (1995) Wavelets and Statistics, Lecture Notes in Statistics. In: Antoniadis, A. and Oppenheim, G., Eds., *Translation-Invariant de-Noiseing*, Springer, Berlin, 125-150. [https://doi.org/10.1007/978-1-4612-2544-7\\_9](https://doi.org/10.1007/978-1-4612-2544-7_9)

- [69] Crouse, M.S., Nowak, R.D. and Baraniuk, R.G. (1998) Wavelet-Based Statistical Signal Processing Using Hidden Markov Models. *IEEE Transactions on Signal Processing*, **46**, 886-902. <https://doi.org/10.1109/78.668544>
- [70] Krim, H., Tucker, D., Mallat, S. and Donoho, D. (1999) On Denoising and Best Signal Representation. *IEEE Transactions on Information Theory*, **45**, 2225-2238. <https://doi.org/10.1109/18.796365>
- [71] Hyvarinen, A., Hoyer, P. and Oja, E. (1998) Sparse Code Shrinkage for Image Denoising. 1998 *IEEE International Joint Conference on Neural Networks Proceedings. IEEE World Congress on Computational Intelligence*, Anchorage, 4-9 May 1998, 859-864.
- [72] Deledalle, C.-A., Salmon, J., Dalalyan, A.S. and Champs-sur-Marne, F. (2011) Image Denoising with Patch Based PCA: Local versus Global. *Proceedings of the British Machine Vision Conference*, Dundee, 29 August-2 September 2011, 1-10. <https://doi.org/10.5244/C.25.25>
- [73] Salmon, J., Harmany, Z., Deledalle, C.-A. and Willett, R. (2014) Poisson Noise Reduction with Non-Local PCA. *Journal of Mathematical Imaging and Vision*, **48**, 279-294. <https://doi.org/10.1007/s10851-013-0435-6>
- [74] Tasdizen, T. (2008) Principal Components for Non-Local Means Image Denoising. *15th IEEE International Conference on Image Processing*, San Diego, 12-15 October 2008, 1728-1731. <https://doi.org/10.1109/ICIP.2008.4712108>
- [75] Abrahamsen, T.J. and Hansen, L.K. (2011) A Cure for Variance Inflation in High Dimensional Kernel Principal Component Analysis. *The Journal of Machine Learning Research*, **12**, 2027-2044.
- [76] Thomas, J.K., Scharf, L.L. and Tufts, D.W. (1995) The Probability of a Subspace Swap in the SVD. *IEEE Transactions on Signal Processing*, **43**, 730-736. <https://doi.org/10.1109/78.370627>
- [77] Salih, M.E., Zhang, X.M. and Ding, M.Y. (2013) An Improvement of Non-Local Means Denoising Method in the Presence of Large Noise. *Applied Mechanics and Materials*, **263**, 223-226. <https://doi.org/10.4028/www.scientific.net/AMM.263-266.223>
- [78] Salih, M.E., Zhang, X. and Ding, M. (2013) Two Modifications of Weight Calculation of the Non-Local Means Denoising Method. *Engineering*, **5**, 522. <https://doi.org/10.4236/eng.2013.510B107>
- [79] Schölkopf, B. and Smola, A.J. (2001) Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond (Adaptive Computation and Machine Learning).
- [80] Turk, M.A. and Pentland, A.P. (1991) Face Recognition Using Eigenfaces. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Maui, 3-6 June 1991, 586-591.
- [81] Martin, S. (2006) The Numerical Stability of Kernel Methods. ISAIM.
- [82] Wu, J., Wang, J. and Liu, L. (2007) Feature Extraction via KPCA for Classification of Gait Patterns. *Human Movement Science*, **26**, 393-411. <https://doi.org/10.1016/j.humov.2007.01.015>
- [83] Nguyen, M.H. and De la Torre, F. (2008) Robust Kernel Principal Component Analysis. *Conference on Neural Information Processing Systems*, Vancouver, 8-11 December 2008, 8 p. <https://proceedings.neurips.cc/paper/2008/file/8f53295a73878494e9bc8dd6c3c7104f-Paper.pdf>

- [84] Mika, S., Schölkopf, B., Smola, A.J., Müller, K.-R., Scholz, M. and Rätsch, G. (1998) Kernel PCA and De-Noising in Feature Spaces. *Conference on Neural Information Processing Systems*, Denver, CO, 536-542.
- [85] Hamprecht, P.F. (2012) Pattern Recognition Class (Nonlinear SVM).
- [86] Kim, K.I., Franz, M.O. and Scholkopf, B. (2005) Iterative Kernel Principal Component Analysis for Image Modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**, 1351-1366. <https://doi.org/10.1109/TPAMI.2005.181>
- [87] Schölkopf, B., Smola, A. and Müller, K.-R. (1998) Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, **10**, 1299-1319. <https://doi.org/10.1162/089976698300017467>
- [88] Jensen, J.A. (1996) Field: A Program for Simulating Ultrasound Systems. *10th Nordic-Baltic Conference on Biomedical Imaging*, Vol. 4, 351-353.
- [89] Bernardes, R., Maduro, C., Serranho, P., Araújo, A., Barbeiro, S. and Cunha-Vaz, J. (2010) Improved Adaptive Complex Diffusion Despeckling Filter. *Optics Express*, **18**, 24048-24059. <https://doi.org/10.1364/OE.18.024048>
- [90] Zhang, F., Yoo, Y.M., Koh, L.M. and Kim, Y. (2007) Nonlinear Diffusion in Laplacian Pyramid Domain for Ultrasonic Speckle Reduction. *IEEE Transactions on Medical Imaging*, **26**, 200-211. <https://doi.org/10.1109/TMI.2006.889735>
- [91] Ullom, J.S., Oelze, M.L. and Sanchez, J.R. (2012) Speckle Reduction for Ultrasonic Imaging Using Frequency Compounding and Despeckling Filters along with Coded Excitation and Pulse Compression. *Advances in Acoustics and Vibration*, **2012**, Article ID: 474039. <https://doi.org/10.1155/2012/474039>
- [92] Sprawls, P. (1993) The Physical Principles of Medical Imaging. 2nd Edition, SPRAWLS Education Foundation, Madison, Wisconsin. <http://www.sprawls.org/ppmi2/BLUR>
- [93] Zhong, H., Li, Y. and Jiao, L. (2011) SAR Image Despeckling Using Bayesian Non-local Means Filter with Sigma Preselection. *Geoscience and Remote Sensing Letters*, **8**, 809-813. <https://doi.org/10.1109/LGRS.2011.2112331>
- [94] Schölkopf, B. (2007) Introduction to Kernel Methods. Max Planck Institute for Biological Cybernetics, Tübingen.
- [95] Kim, K.I., Franz, M. and Schölkopf, B. (2004) Kernel Hebbian Algorithm for Single-Frame Super-Resolution.
- [96] Field, D.J. (1994) What Is the Goal of Sensory Coding? *Neural Computation*, **6**, 559-601. <https://doi.org/10.1162/neco.1994.6.4.559>
- [97] Kim, K.I., Franz, M.O. and Schölkopf, B. (2003) Kernel Hebbian Algorithm for Iterative Kernel Principal Component Analysis. Max-Planck-Institut für biologische Kybernetik, Tübingen, Tech. Rep. 109.
- [98] Hiremath, P. and Prabhakar, C. (2006) Acquiring Non Linear Subspace for Face Recognition Using Symbolic Kernel PCA Method. *Journal of Symbolic Data Analysis*, **4**, 15-26. [https://doi.org/10.1142/9789812772381\\_0008](https://doi.org/10.1142/9789812772381_0008)
- [99] Yang, M.-H., Ahuja, N. and Kriegman, D. (2000) Face Recognition Using Kernel Eigenfaces. 2000 *IEEE International Conference on Image Processing*, Vancouver, 10-13 September 2000, 37-40.
- [100] Günter, S., Schraudolph, N.N. and Vishwanathan, S. (2007) Fast Iterative Kernel Principal Component Analysis. *Journal of Machine Learning Research*, **8**, 1893-1918.
- [101] Jackson, J.E. (2005) A User's Guide to Principal Components. John Wiley & Sons, Hoboken.



- [102] Garcia, E. (2006, September) SVD and LSI Tutorial 3: Computing the Full SVD of a Matrix.  
<https://cs.fit.edu/~dmitra/SciComp/Resources/singular-value-decomposition-fast-track-tutorial.pdf>
- [103] Autumn, P.A.N. (2009) Lecture Machine Learning (CS 229).
- [104] Abu-Mostafa, P.Y. (2012) Caltech's Machine Learning Course.

## Appendix A. Mathematical Background

The PCA is the main linear technique for dimension reduction, which is used in various areas applications such as data visualization and image compression, noise reduction [101]. The philosophy of data reduction, given data points in  $n$ -dimensional space is to project the data point into lower dimensional space while preserving as much information as possible. The SVD is an efficient algorithm to perform the PCA. It is different from eigenvector calculating thing, and it is available in Matlab. Here, we detail the principles of SVD. Then, we give the definition of the kernel method and how to extract the significant dimensions of the data set. Finally, we describe how to calculate the kernel SVD high dimensional feature using the kernel theory.

### A.1. Singular Value Decomposition

In this section, steps of how to calculate the SVD are presented. Each step is expressed mathematically, in addition to the corresponding Matlab command.

Given a database  $X = \{x_1, \dots, x_m\}$ , consist of  $m$   $x_i$  features, (for example, a concatenation of

Given matrix  $X \in \mathbb{R}^{m \times n}$ , SVD consist of the product of three matrices  $\Sigma$ ,  $U$ , and  $V$ :

$$X = U\Sigma V^T \quad (\text{A-1})$$

where:

$\Sigma \in \mathbb{R}^{m \times n}$  is a diagonal matrix with positive singular values  $\sigma_i$  in the diagonal, regard that singular values go form largest to smallest:  $\sigma_1 > \sigma_2 > \dots > \sigma_n$ .

$U \in \mathbb{R}^{m \times m}$  is the left eigenvector matrix, and it has an orthonormal columns.

$V^T \in \mathbb{R}^{n \times n}$  is the right eigenvector matrix, and has an orthonormal rows.

$X^T X$  and  $XX^T$  are symmetric matrices by definition

$$(XX^T)^T = X^T X^T T = X^T X \quad (\text{A-2})$$

Computing the eigenvector of  $XX^T$  and  $X^T X$  and is the way to estimate  $U$ , and  $V$  matrices using the definition of SVD in, respectively. While  $\Sigma$  is constructed by estimating the positive square roots of the of the nonzero eigenvalues of  $X^T X$ , or  $XX^T$

$$X^T X = (U\Sigma V^T)^T U\Sigma V^T = (V^T \Sigma^T U^T) U\Sigma V^T = V\Sigma U^T U\Sigma V^T = V\Sigma^2 V^T \quad (\text{A-3})$$

$$XX^T = U\Sigma V^T (U\Sigma V^T)^T = U\Sigma V^T (V^T \Sigma^T U^T) = U\Sigma V^T V\Sigma U^T = U\Sigma^2 U^T \quad (\text{A-4})$$

We have  $m$  singular vectors (instead of eigenvector). The left singular vectors have size  $m$ , and the right singular vector have size  $n$ .

One of the SVD properties is to produce orthogonal matrices.  $V$  is orthogonal, *i.e.*  $V^T = V^{-1}$ ,  $VV^T = I$ , where  $I$  is the identity matrix. But, it is not true that  $UU^T = I$ , and so we do not called  $U$  an orthogonal matrix. Some code adds some extra artificial columns to  $U$  matrix and zeroes  $\Sigma$  matrix for dimension

match. This it does by a process called (Gram–Schmidt process), then  $U$  will have orthonormal columns, and  $U^T U$  is just the identity matrix.  $\Sigma$  is a diagonal matrix, thus it has zeroes off diagonal everywhere. It is  $n \times n$  (where  $n$  is the original dimension of the data). Each value in  $\Sigma$  is an eigenvalue for one of the eigenvectors.

In summary, the SVD essentially give the PC's in  $V^T$ , and it gives you each of their eigenvalues in  $\Sigma$ . Most of implementation of SVD actually goes to the extra step of sorting the rows of the  $V^T$  and  $\Sigma$ , so the eigenvectors values goes from largest to smallest.

$U$ ,  $\Sigma$ , and  $V$  matrices are generated in Matlab by:

$$[U, \Sigma, V] = \text{svd}(X).$$

The statistical property of the SVD is based on representation of data in subspaces of significant [16]. This is core property to derive the less noise image features. The lower dimension representation of the data can be obtained by using a truncated SVD to get a compressed version of the data matrix. This truncation can do by keeping the first  $d$  singular values. Using form I, the reduction of  $X$  is given as

$$X^d = U^d \Sigma^d V^{dT} \quad (\text{A-5})$$

where  $X^d$  is referred to as the rank  $d$  approximation of  $X$ , or the “Reduced SVD” of  $X$ . For example, If we eliminate dimensions by keeping the three largest singular values, this is a rank 3 approximation [102]. Now are going to look how to use SVD to do some cool stuff in data processing.

## A.2. Kernel Method

### A.2.1. Transformation between Distance and Similarity Measure

The measure of distance is an important routine in data processing and analysis. One of the mostly used dissimilarity measure is the Euclidean distance. It is defined as the  $L_2$  norm (square root of the vector inner product) of the difference of the two vectors or two points. If the similarity is interpreted as a covariance, then is the Euclidean distance could be written as a similarity matrix.

$$d_{i,j}^2 = \|y_i - y_j\|_2^2 = \langle (y_i - y_j)^2 \rangle = (y_i - y_j)(y_i - y_j)^T = y_i y_i^T + y_j^T y_j - 2y_i y_j^T \quad (\text{A-6})$$

If the covariance is of the form:

$$\begin{aligned} k_{i,j} &= y_{i,:}^T y_{j,:} \text{ then} \\ d_{i,j}^2 &= k_{i,i} + k_{j,j} - 2k_{i,j} \\ d_{i,j}^2 &= \begin{cases} 0 & \text{if } y_i = y_j \\ 2 & \text{if } y_i \neq y_j \end{cases} \end{aligned} \quad (\text{A-7})$$

So a concept called kernel rises up, which it considered as a transformation between distance and similarity matrix, and it becomes a basic for a number of algorithms in machine learning algorithms. An explanation of the kernel method theorem is introduced in the following section.

**A.2.2. Kernel Similarity Measurement**

Kernel method comes up with a different idea for similarity measurement. And the difference is that, the kernel calculates distance in the space of transformed feature. Given the transformation ( $\Phi$ ), that maps the data from original feature space to some higher dimensional feature space. As shown in the **Figure A-1**,  $\Phi$  takes points  $x_i$  and  $x_j$  mapped them into a Gaussian centered on  $x_i$  and  $x_j$  respectively.

$$\begin{aligned}
 \text{Feature map } (\Phi) \quad \Phi : X &\rightarrow \mathbb{R}^X, x \mapsto k(., x) . \\
 \Phi : X &\mapsto H \\
 k(x_i, x_j) &= \langle \Phi(x_i), \Phi(x_j) \rangle
 \end{aligned}
 \tag{A-8}$$

Distance in transformed feature space is computed as the following:

$$\begin{aligned}
 \|\Phi(x_i) - \Phi(x_j)\|_2^2 &= \langle \Phi(x_i) - \Phi(x_j), \Phi(x_i) - \Phi(x_j) \rangle \\
 &= \langle \Phi(x_i), \Phi(x_i) \rangle - 2\langle \Phi(x_i), \Phi(x_j) \rangle + \langle \Phi(x_j), \Phi(x_j) \rangle \\
 &= k(x_i, x_i) - 2k(x_i, x_j) + k(x_j, x_j)
 \end{aligned}
 \tag{A-9}$$

The kernel is the same as a dot product of mapped features. And in a sense of similarity measurement the kernel function is a symmetric function that maps a pair features and give a real number. The kernel gives large number value if the two inputs are similar, whereas in contrast low value if the inputs are dissimilar.

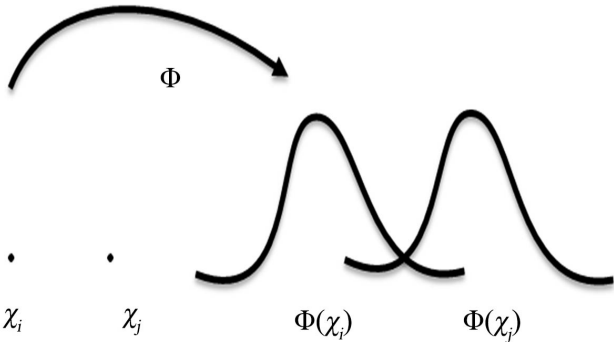
**A.2.3. Kernel and Reproducing Kernel Hilbert Space (RKHS)**

Riesz’ representation theorem tells whenever, there is a linear continuous function ( $f$ ) it can be represented as a dot product with other some element of Hilbert space ( $H$ ). The  $H$  can be defined as an inner product space that is complete and separable with respect to the norm defined by the inner product.

The Riesz’ representation theorem states that, there is an element  $r_x \in H$  can be written as:

$$\langle r_x, f \rangle = f(x)
 \tag{A-10}$$

Using Riesz’ representation theorem the kernel can be defined as a reproducing kernel Hilbert space (RKHS) if:



**Figure A1.** Graphical illustration of the feature space of the Gaussian kernel, reprinted from [79], p. 32.

$$\text{span}\left(\{\Phi(x) : x \in X\}\right) = \left\{ f(\cdot) = \sum_i \alpha_i k(\cdot, x_i) : x_i \in X, \alpha_i \in \mathbb{R} \right\} \quad (\text{A-11})$$

Given kernel  $k : X \times X \rightarrow \mathbb{R}$ , one can construct the RKHS as the completion of the space of functions spanned by the set with an inner product defined as follows.

Consider:

$$f(\cdot) = \sum_i \alpha_i k(\cdot, x_i) \quad (\text{A-12})$$

$$g(\cdot) = \sum_j \beta_j k(\cdot, x_j) \quad (\text{A-13})$$

$$\langle f, g \rangle = \sum_{i,j} \beta_j \alpha_i k(x_j, x_i) \quad (\text{A-14})$$

Note that  $\langle f, k(\cdot, x) \rangle = \sum_i \alpha_i k(x, x_i) = f(x)$  ( $k$  has the reproducing property).

**Testing** that  $\langle f, g \rangle$  is an inner product is by checking the following conditions:

1) Symmetry

$$\langle f, g \rangle = \sum_{i,j} \alpha_i \beta_j k(x_i, x_j) = \sum_{i,j} \beta_j \alpha_i k(x_j, x_i) = \langle g, f \rangle$$

2) Positive definiteness

$$\langle f, f \rangle = \alpha K \alpha \geq 0$$

$$\left\langle \sum_i \alpha_i k(\cdot, x_i), \sum_i \alpha_i k(\cdot, x_i) \right\rangle = \sum_{ij} \alpha_i \alpha_j k(x_i, x_j) \geq 0 \quad (\text{A-15})$$

So as long as we define a kernel function and construct the kernel matrix and it is positive definite kernel. It means we could find a mapping such that it is possible to rewrite the kernel function in term of inner product of the mapped features. Conversely, for every RKHS there exists an associated reproducing kernel which is symmetric and positive definite (PD). The Mercer kernels is the family of kernel functions for which the kernel trick is guaranteed to work independent of the data you have looking at.

#### A.2.4. Kernel Trick

if  $\Phi(x)$  is an extremely high dimensional, constructing the kernel need to represent the extremely high dimensional feature vector and then computing the inner products in the feature space which seem computationally inefficient and very expensive. However, by using kernel trick, it is possible to compute the inner product between these two vectors very inexpensively, and easily working in feature spaces even if they are very high dimensional. So for computing the distances, if given a kernel function, the need is just only evaluating the kernel and gets a value for the transformed inner product without telling how  $\Phi(x)$  basically look like. That means there is no need to think about  $\Phi(x)$  and build inner product but just about the kernel. From the computation point of view, the evaluation of the kernel function is much easier than the computation of the

transformation of the feature followed by the inner product computation which is more complex. So just by evaluating the kernel  $k(x_i, x_j)$  and knowing that there is a map and inner product). An illustration of the computation expense of computing dot product in the feature space, and basic idea of kernel trick is given in the following example. The example shows that the inner products in the feature space could be evaluated implicitly in the input space. Assuming there is a transformation mapping from original two dimensions features to some higher three dimensional set of features

$$\begin{aligned} \Phi : \mathbb{R}^2 &\rightarrow \mathbb{R}^3 \\ (x_1, x_2) &\mapsto (z_1, z_2, z_3) := (x_1^2 + \sqrt{2}x_1x_2 + x_2^2) \end{aligned} \tag{A-16}$$

where  $\Phi(x)$  is represented in term quadratic form (all possible products of pairs of the components of the variable  $x$ ).

In this case ( $n = 2$ ) is a dimension of  $x$ , the length of  $\Phi(x)$  is  $n^2$

- $O(n^2)$  is needed just to compute  $\Phi(x)$

$$\Phi(x) = \begin{pmatrix} x_1x_1 \\ x_1x_2 \\ x_2x_1 \\ x_2x_2 \end{pmatrix} \approx \begin{pmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{pmatrix} \tag{A-17}$$

- Then  $O(n)$  is needed to compute the kernel which is inner product in the feature space.

$$\begin{aligned} k(x_i, x_j) &= \langle \Phi(x_i), \Phi(x_j) \rangle = (x_i^2, \sqrt{2}x_i x_{i_2}, x_{i_2}^2) (x_j^2, \sqrt{2}x_j x_{j_2}, x_{j_2}^2)^T \\ &= \langle x_i, x_j \rangle^2 = x_i^2 x_j^2 + 2x_i x_{i_2} x_j x_{j_2} + x_{i_2}^2 x_{j_2}^2 \\ &= (x_i x_{j_1})^2 + 2(x_i x_{j_1} x_{i_2} x_{j_2}) + (x_{i_2} x_{j_2})^2 \end{aligned} \tag{A-18}$$

The inner product in the feature space can evaluate in the input space, because of the followings:

- The  $x_i x_{j_1}$  and  $x_{i_2} x_{j_2}$  are the dot product terms taken in the input space.
- The  $(x_i x_{j_1} + x_{i_2} x_{j_2})^2$  is the dot product term taken in the input space raised to power of 2.

As it is mentioned before that the kernel function has the property that it correspond to dot product in some other representation rather than the input space, and here the other representation is simply the representation computing the all product of order 2. And that it means the dot product in  $H$  can be computed in  $R^2$ . The example showed that the kernel trick is very computationally efficient  $k(x_i, x_j)$  is defined as  $\langle x_i, x_j \rangle^2$ , i.e. it is just take the inner product between  $x_i$  and  $x_j$  which is  $O(n)$ , then square that and the kernel function is computed, and implicitly working in an extremely high dimensional computing feature space [103]. In the above example examined only the 2D case, but the  $n$  dimensional case is just generalization of the 2D case.

$$\begin{aligned} x_i, x_j &\in \mathbb{R}^n \\ \langle x_i, x_j \rangle^r &= \left( \sum_{p=1}^n x_{i,p} x_{j,p} \right)^r = \langle \Phi(x_i), \Phi(x_j) \rangle^r \end{aligned} \tag{A-19}$$

So you have  $n$  dimensional  $x_i$  and  $x_j$  space vectors, then calculate the dot product (get a single number) and raise it to power of  $r$ . So we get a simple summation operation, and it does not matter if the  $r$  is big or large, since we get the same computation complexity [104].

### A.2.5. Kernel Function

The typical kernel functions that can express the similarity between the  $x_i$  and  $x_j$  are:

- Linear: (*i.e.* there is no transformation, but just it is computing the inner product of two input vectors)

$$k(x_i, x_j) = \langle x_i, x_j \rangle \quad (\text{A-20})$$

- Gaussian radial basis function

$$k(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|_2^2}{\sigma^2}\right) \quad (\text{A-21})$$

It is considered as one of the preferred kernel function, which computes the Gaussian with square distance between  $x_i$  and  $x_j$ . It takes the points and mapped them into a Gaussian function centered on the  $x_i$  and  $x_j$  points as shown in the **Figure A1**. In other words, each point is represented by its similarity to all other points [79]. This means that a high dimensional vector can be obtained by evaluating the kernel function on these finite set of points.

### A.3. Support Vector Machine

This section addresses one of the main applications of the kernel method which explain why kernel method could be useful. The support vector machine (SVM) is one of the powerful classification algorithms, which looking for a decision surface that separates between the two groups of the data points. These points (for example faces and non faces data points) are referred to as the support vectors. **Figure A2(b)** shows separable training data sets, it seem to it is impossible to use a linear separator. Thus a more complicated (curve instead of line) nonlinear classifier is needed. Applying the kernel trick is a way to create kernel based SVM classifiers, and this allows the algorithm to separate the data points using a hyper plane in a transformed feature space.

Consider a map into a feature space (higher dimensional, 3D) the new coordinates are product of old two coordinate.

$$\begin{aligned} \Phi: \mathbb{R}^2 &\rightarrow \mathbb{R}^3 \\ (x_1, x_2) &\mapsto (z_1, z_2, z_3) := (x_1^2 + \sqrt{2}x_1x_2 + x_2^2) \end{aligned} \quad (\text{A-22})$$

So there is some amount of red data points appear close to the mean in the 2D representation **Figure A2(a)**. But they are actually sitting away from the mean in 3D space.

This means if the high dimensional data are not visualized, then the some of

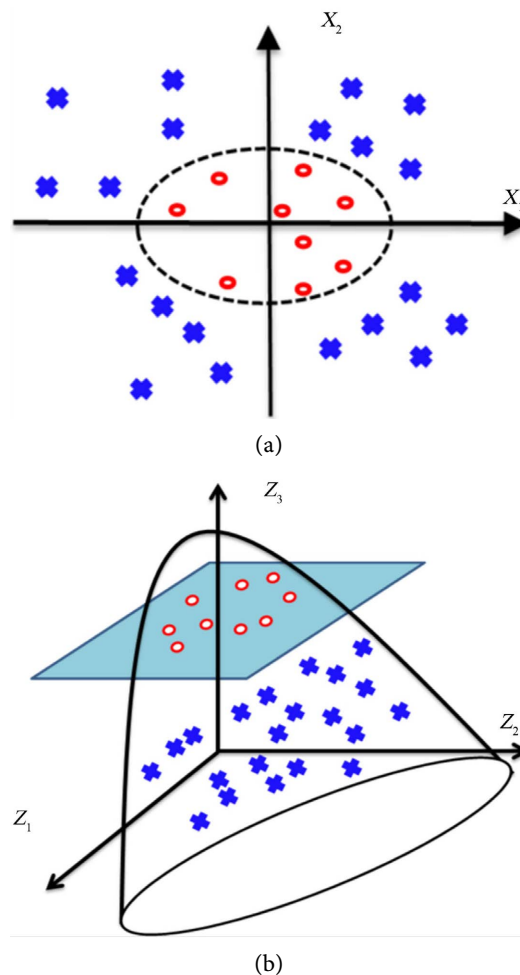
the data is seem to projecting down to the mean. The cross and circle data points are not linearly separable in the original input space, and using kernel they can be linearly separable in 3D high-dimensional feature space. See **Figure A2(b)**

#### A.4. Kernel Principal Component Analysis

As mentioned above, that SVD is a way to do the eigenvalue problem by matrix factorization. The kernel method comes up with a different idea, that the kernel represents a dot product in higher dimension space using some nonlinear transformation of the input matrix [81]. A description of the kernel SVD algorithm is presented in this section.

$$X^z = U^z \Sigma^z V^{zT} \quad (\text{A-23})$$

$$X^z = \Phi(X) \quad (\text{A-24})$$



**Figure A2.** The idea of kernel based SVM classifier. Applying mapping into higher dimensional feature space using the kernel trick is a way to construct a separating hyperplane there (b). This is equivalent to the nonlinear decision boundary in input space (a). Adapted from [79], p. 29.



So computing the eigenvector of  $X^z X^{z^T}$  and  $X^{z^T} X^z$  is the way to estimate  $U^z, \Sigma^z, V^{z^T}$  matrices, respectively using the definition of SVD in (A-3) and (A-4). The computation of  $X^z X^{z^T}$  or  $X^{z^T} X^z$  (the inner product of transformed matrix) can be implemented implicitly in the input space using kernel trick.

$$K = X^z X^{z^T} = U^z \Sigma^{z^2} V^{z^T} \quad (\text{A-25})$$

$$K = X^{z^T} X^z = V^z \Sigma^{z^2} V^{z^T} \quad (\text{A-26})$$

Kernel SVD does the decomposition in terms of inner product of the transformed feature, which modeled by a kernel function. In other words the inner products  $X^z X^{z^T}$  and  $X^{z^T} X^z$  are replaced by kernels to yield a nonlinear version of SVD. And the decomposition of  $K$  will produce the  $U^z, \Sigma^z, V^{z^T}$  matrices which have properties similar to the properties of  $U, \Sigma, V^T$  matrices.

$$K = U^h \Sigma^h V^{h^T} \quad (\text{A-27})$$