Edge Preserving Image Denoising in Reproducing Kernel Hilbert Spaces

A novel approach for removing any type of additive noise from a grayscale image

P. Bouboulis[1] K. Slavakis[2], S. Theodoridis[1]

[1]Department of Informatics and Telecommunications
University of Athens Greece,
[2]Department of Telecommunications, Science and Technology University of Peloponnese, Greece

25-08-2010

Outline

1. Image Denoising
   - The problem
   - Typical Solutions
2. Reproducing Kernel Hilbert Spaces
   - Definition and Main Properties of RKHS
   - Why RKHS?
   - Two Important Theorems
3. Kernelised Noise Removal
   - Basic Idea
   - Formulation
   - Parameter Selection
   - Experiments

Outline

1. Image Denoising
   - The problem
   - Typical Solutions
2. Reproducing Kernel Hilbert Spaces
   - Definition and Main Properties of RKHS
   - Why RKHS?
   - Two Important Theorems
3. Kernelised Noise Removal
   - Basic Idea
   - Formulation
   - Parameter Selection
   - Experiments

Additive Noise

- f: is the original image.

Additive Noise

- f: is the original image.
- e: is the additive noise.

Additive Noise

- f: is the original image.
- e: is the additive noise.
- $\hat{f} = f + e$: is the noisy image.

Additive Noise

- f: is the original image.
- e: is the additive noise.
- $\hat{f} = f + e$: is the noisy image.
- The objective of the image denoising problem is to estimate the original image f from the noisy one $\hat{f}$.

Additive Noise

Additive Noise

Types of Noise

Types of Noise that we typically encounter:

Types of Noise

Types of Noise that we typically encounter:

1. Gaussian noise: $p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\mu)^2/(2\sigma^2)}$

Types of Noise

Types of Noise that we typically encounter:

1. Gaussian noise: $p(z) = \frac{1}{\sqrt{2\pi\sigma}} e^{-(z-\mu)^2/(2\sigma^2)}$
2. Impulse Noise: $p(z) = P_a$, if $z = a$, $p(z) = P_b$, if $z = b$, $p(z) = 0$ otherwise.

Outline

Typical Solutions

Typical Solutions

Typical Solutions

Typical Solutions

- Median Filter

Typical Solutions

Typical Solutions

- Median Filter
- Fourier Analysis

Typical Solutions

Typical Solutions

- Median Filter
- Fourier Analysis
- Wavelets

Typical Solutions

Typical Solutions

- Median Filter
- Fourier Analysis
- Wavelets
- Partial Differential Equations

Problems and Motivation

- Most known methods are noise specific, i.e., they need some information about the type and/or the statistics of the noise (e.g., the parameter σ in the case of gaussian noise).

Problems and Motivation

- Most known methods are noise specific, i.e., they need some information about the type and/or the statistics of the noise (e.g., the parameter σ in the case of gaussian noise).
- We aim at a noise independent methodology.

Problems and Motivation

- Most known methods are noise specific, i.e., they need some information about the type and/or the statistics of the noise (e.g., the parameter σ in the case of gaussian noise).
- We aim at a noise independent methodology.
- The idea is to express f as a span of some base functions f_i.

Problems and Motivation

- Most known methods are noise specific, i.e., they need some information about the type and/or the statistics of the noise (e.g., the parameter σ in the case of gaussian noise).
- We aim at a noise independent methodology.
- The idea is to express f as a span of some base functions f_i.
- We choose the base functions f_i to belong to a RKHS.

Outline

1. Image Denoising
   - The problem
   - Typical Solutions
2. Reproducing Kernel Hilbert Spaces
   - Definition and Main Properties of RKHS
   - Why RKHS?
   - Two Important Theorems
3. Kernelised Noise Removal
   - Basic Idea
   - Formulation
   - Parameter Selection
   - Experiments

Reproducing Kernel Hilbert Spaces.

Consider a linear class $\mathcal{H}$ of real valued functions f defined on a set $\mathcal{X}$ (in particular $\mathcal{H}$ is a Hilbert space) for which there exists a function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ with the following two properties:

Reproducing Kernel Hilbert Spaces.

Consider a linear class $\mathcal{H}$ of real valued functions f defined on a set $\mathcal{X}$ (in particular $\mathcal{H}$ is a Hilbert space) for which there exists a function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ with the following two properties:

1. For every $x \in \mathcal{X}$, $\kappa(x, \cdot)$ belongs to $\mathcal{H}$.

Reproducing Kernel Hilbert Spaces.

Consider a linear class $\mathcal{H}$ of real valued functions f defined on a set $\mathcal{X}$ (in particular $\mathcal{H}$ is a Hilbert space) for which there exists a function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ with the following two properties:

1. For every $x \in \mathcal{X}$, $\kappa(x, \cdot)$ belongs to $\mathcal{H}$.
2. κ has the so called reproducing property, i.e.,

$$f(x) = \langle f, \kappa(x, \cdot) \rangle_{\mathcal{H}}, \text{ for all } f \in \mathcal{H}, \; x \in \mathcal{X}, \tag{1}$$

in particular $\kappa(x, y) = \langle \kappa(x, \cdot), \kappa(y, \cdot) \rangle_{\mathcal{H}}$.

Outline

Non linear Processing in RKHS

- Why are RKHS so useful?

Non linear Processing in RKHS

- Why are RKHS so useful?
- The original nonlinear task is transformed into a linear one, which can be solved by employing an easier algebra.

Non linear Processing in RKHS

- Why are RKHS so useful?
- The original nonlinear task is transformed into a linear one, which can be solved by employing an easier algebra.
- The main concepts of this procedure can be summarized in the following two steps:

Non linear Processing in RKHS

- Why are RKHS so useful?
- The original nonlinear task is transformed into a linear one, which can be solved by employing an easier algebra.
- The main concepts of this procedure can be summarized in the following two steps:
  1. Map the finite dimensionality input data from the input space $\mathcal{X}$ into a higher dimensionality (possibly infinite) RKHS $\mathcal{H}$.

Non linear Processing in RKHS

- Why are RKHS so useful?
- The original nonlinear task is transformed into a linear one, which can be solved by employing an easier algebra.
- The main concepts of this procedure can be summarized in the following two steps:
  1. Map the finite dimensionality input data from the input space $\mathcal{X}$ into a higher dimensionality (possibly infinite) RKHS $\mathcal{H}$.
  2. Perform a linear processing on the mapped data in $\mathcal{H}$.

The Kernel Trick

- An alternative way of describing this process is through the popular kernel trick.

The Kernel Trick

- An alternative way of describing this process is through the popular kernel trick.
- "Given an algorithm which is formulated in terms of an inner product, one can construct an alternative algorithm by replacing the inner product with a positive kernel κ".

Some Kernels used in practice

- **Polynomial Kernel** $\kappa(x, y) = \langle x, y \rangle^d$

Some Kernels used in practice

- **Polynomial Kernel** $\kappa(x, y) = \langle x, y \rangle^d$
- **Gaussian Kernel** $\kappa(x, y) = exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$, $\sigma > 0$

Some Kernels used in practice

- **Polynomial Kernel** $\kappa(x, y) = \langle x, y \rangle^d$
- **Gaussian Kernel** $\kappa(x, y) = exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$, $\sigma > 0$
- **Inhomogeneous Polynomial Kernel** $\kappa(x, y) = (\langle x, y \rangle + c)^d$

Some Kernels used in practice

- **Polynomial Kernel** $\kappa(x, y) = \langle x, y \rangle^d$
- **Gaussian Kernel** $\kappa(x, y) = exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$, $\sigma > 0$
- **Inhomogeneous Polynomial Kernel** $\kappa(x, y) = (\langle x, y \rangle + c)^d$
- **B_n-Spline of odd order Kernel** $\kappa(x, y) = B_{2r+1}(\|x - y\|)$, with $B_n = \bigotimes_{i=1}^{n} I_{[-\frac{1}{2},\frac{1}{2}]}$

Outline

The Representer Theorem

Theorem

Denote by $\Omega : [0, \infty) \to \mathbb{R}$ *a strictly monotonic increasing function, by* $\mathcal{X}$ *a set and by* $c : (\mathcal{X} \times \mathbb{R}^2)^m \to \mathbb{R} \cup \{\infty\}$ *an arbitrary loss function. Then each minimizer* $f \in \mathcal{H}$ *of the regularized risk functional*

$$c\left((x_1, y_1, f(x_1)), \ldots, (x_N, y_N, f(x_N)\right) \quad + \quad \Omega\left(\|f\|_{\mathcal{H}}\right)$$

admits a representation of the form

$$f(x) = \sum_{n=1}^{N} \alpha_n \kappa(x_n, x).$$

Example of the Representer Theorem

Consider the problems

$$\underset{f\in\mathcal{H}}{\text{minimize}} \quad \sum_{n=1}^{N} |f(x_i) - y_i|^2 + \lambda \|f\|_{\mathcal{H}}^2$$

Example of the Representer Theorem

Consider the problems

$$\underset{f\in\mathcal{H}}{\text{minimize}} \quad \sum_{n=1}^{N} |f(x_i) - y_i|^2 + \lambda\|f\|_{\mathcal{H}}^2$$

$$\underset{f\in\mathcal{H}}{\text{minimize}} \quad \sum_{n=1}^{N} |f(x_i) - y_i| + \lambda\|f\|_{\mathcal{H}}^2$$

Example of the Representer Theorem

Consider the problems

$$\operatorname*{minimize}_{f\in\mathcal{H}} \sum_{n=1}^{N} |f(x_i) - y_i|^2 + \lambda \|f\|_{\mathcal{H}}^2$$

$$\operatorname*{minimize}_{f\in\mathcal{H}} \sum_{n=1}^{N} |f(x_i) - y_i| + \lambda \|f\|_{\mathcal{H}}^2$$

In both cases the minimizer admits the form:

$$f(x) = \sum_{n=1}^{N} \alpha_n \kappa(x_n, x).$$

The semi-parametric Representer Theorem

Theorem

Suppose that in addition to the assumptions of the previous theorem we are given a set of M real-valued functions $\{\psi_p\}_{p=1}^M : \mathcal{X} \to \mathbb{R}$, with the property that the $N \times M$ matrix $(\psi_p(x_n))_{n,p}$ has rank M. Then any $f := \tilde{f} + h$, with $\tilde{f} \in \mathcal{H}$ and $h \in span\{\psi_p\}$, minimizing the regularized risk functional

$$c\left((x_1, y_1, f(x_1)), \ldots, (x_N, y_N, f(x_N)\right) \quad + \quad \Omega\left(\|\tilde{f}\|_{\mathcal{H}}\right)$$

admits a representation of the form

$$f(x) = \sum_{n=1}^{N} \alpha_n \kappa(x_n, x) + \sum_{p=1}^{M} \beta_p \psi_p(x).$$

The semi-parametric Representer Theorem

- Typically a RKHS consists of functions that are very smooth.

The semi-parametric Representer Theorem

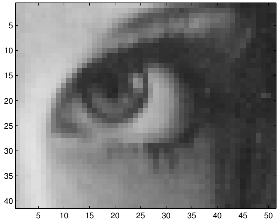
- Typically a RKHS consists of functions that are very smooth.
- Evidently, one cannot effectively approximate a non-smooth function f as a span of base functions of a specific RKHS.

The semi-parametric Representer Theorem

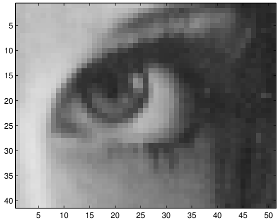
- Typically a RKHS consists of functions that are very smooth.
- Evidently, one cannot effectively approximate a non-smooth function f as a span of base functions of a specific RKHS.
- The semi-parametric Representer Theorem, may be used to impose non-smoothness through the functions ψ_p.

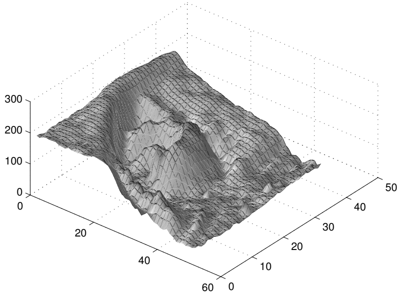
Outline

An image is actually a function



An image is actually a function





Outline

Rectangular area neighborhood

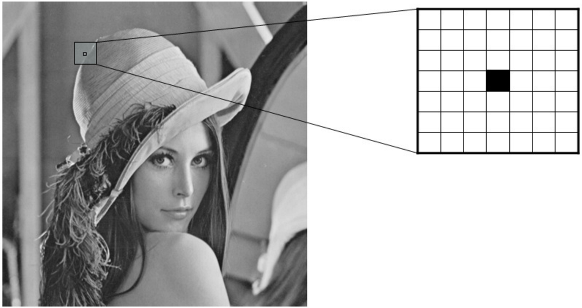
- We are given a noisy image with dimensions $N0 \times M0$

Rectangular area neighborhood

- We are given a noisy image with dimensions $N0 \times M0$
- We move from one pixel to the next taking (for each pixel) a corresponding neighborhood (i.e. a rectangular area).

Rectangular area neighborhood

- We are given a noisy image with dimensions $N0 \times M0$
- We move from one pixel to the next taking (for each pixel) a corresponding neighborhood (i.e. a rectangular area).



Choosing functions to represent edges

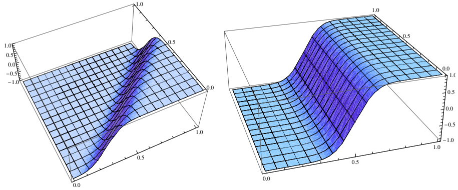
- Let $\hat{f}$ be the given "noisy" neighborhood of one pixel with dimensions $N \times M$, i.e. the $\hat{z}_{m,n} = \hat{f}(x_m, y_n)$ for $m = 1, \ldots, M$, $n = 1, \ldots, N$, are the given pixel values of the noisy neighborhood.

Choosing functions to represent edges

- Let $\hat{f}$ be the given "noisy" neighborhood of one pixel with dimensions $N \times M$, i.e. the $\hat{z}_{m,n} = \hat{f}(x_m, y_n)$ for $m = 1, \ldots, M$, $n = 1, \ldots, N$, are the given pixel values of the noisy neighborhood.
- We assume a set of real valued functions ψ_k, $k = 1, \ldots, K$ defined on $\mathbb{R}^2$ that satisfy the condition of the semiparametric Representer Theorem.

Choosing functions to represent edges

- Let $\hat{f}$ be the given "noisy" neighborhood of one pixel with dimensions $N \times M$, i.e. the $\hat{z}_{m,n} = \hat{f}(x_m, y_n)$ for $m = 1, \dots, M$, $n = 1, \dots, N$, are the given pixel values of the noisy neighborhood.
- We assume a set of real valued functions ψ_k, $k = 1, \dots, K$ defined on $\mathbb{R}^2$ that satisfy the condition of the semiparametric Representer Theorem.



The expansion

- Next, we assume for the denoised image f that $f \in \mathcal{F} = \mathcal{H} \; + \; h_0 \boldsymbol{I} \; + \; \mathrm{span}\{\psi_1, \ldots, \psi_K\}$ (where $\boldsymbol{I} \in \mathbb{R}$ stands for the constant function i.e. $\boldsymbol{I}(x, y) = 1$).

The expansion

- Next, we assume for the denoised image f that
  $f \in \mathcal{F} = \mathcal{H} \ + \ h_0 \boldsymbol{I} \ + \ \mathrm{span}\{\psi_1, \dots, \psi_K\}$
  (where $\boldsymbol{I} \in \mathbb{R}$ stands for the constant function i.e. $\boldsymbol{I}(x, y) = 1$).
- Hence f admits the form

$$f = \tilde{f} + h_0 \boldsymbol{I} + \sum_{k=1}^{K} \beta_k \psi_k.$$

Optimization

- We solve the following minimization problem for each pixel (using Polyak's Projected Subgradient Method):

$$\underset{f \in \mathcal{F}}{\text{minimize}} \quad \sum_{m=1}^{M} \sum_{n=1}^{N} |f(x_m, y_n) - \hat{z}_{m,n}| \quad + \quad \frac{\lambda}{2} \|\tilde{f}\|_{\mathcal{H}}^2 \quad + \quad \frac{\mu}{2} \sum_{k=1}^{K} |\beta_k|^2,$$

where $\tilde{f}$ is the part of the expansion of f that lives on $\mathcal{H}$.

Semiparametric Representer Theorem

Applying a version of the semiparametric Representer Theorem we take that f admits the form

$$f = \sum_{m=1}^{M} \sum_{n=1}^{N} \alpha_{m,n} \kappa\left((x_m, y_n), (\cdot, \cdot)\right) + h_0 \boldsymbol{I} + \sum_{k=1}^{K} \beta_k \psi_k.$$

Remarks

$$\operatorname*{minimize}_{f\in\mathcal{F}} \quad \sum_{m=1}^{M}\sum_{n=1}^{N}|f(x_m, y_n) - \hat{z}_{m,n}| \quad + \quad \frac{\lambda}{2}\|\tilde{f}\|_{\mathcal{H}}^2 \quad + \quad \frac{\mu}{2}\sum_{k=1}^{K}|\beta_k|^2,$$

Remarks

$$\underset{f \in \mathcal{F}}{\text{minimize}} \quad \sum_{m=1}^{M} \sum_{n=1}^{N} |f(x_m, y_n) - \hat{z}_{m,n}| \quad + \quad \frac{\lambda}{2} \|\tilde{f}\|_{\mathcal{H}}^2 \quad + \quad \frac{\mu}{2} \sum_{k=1}^{K} |\beta_k|^2,$$

- Note that the use of l_2 instead of the l_1 norm in the cost function would make the method sensitive to outliers (e.g., impulses).

Remarks

$$\underset{f \in \mathcal{F}}{\text{minimize}} \quad \sum_{m=1}^{M} \sum_{n=1}^{N} |f(x_m, y_n) - \hat{z}_{m,n}| \quad + \quad \frac{\lambda}{2} \|\tilde{f}\|_{\mathcal{H}}^2 \quad + \quad \frac{\mu}{2} \sum_{k=1}^{K} |\beta_k|^2,$$

- Note that the use of l_2 instead of the l_1 norm in the cost function would make the method sensitive to outliers (e.g., impulses).
- Furthermore, the l_1 norm adds some sort of sparsity to the expansion.

Remarks

$$\underset{f \in \mathcal{F}}{\text{minimize}} \quad \sum_{m=1}^{M} \sum_{n=1}^{N} |f(x_m, y_n) - \hat{z}_{m,n}| \quad + \quad \frac{\lambda}{2} \|\tilde{f}\|_{\mathcal{H}}^2 \quad + \quad \frac{\mu}{2} \sum_{k=1}^{K} |\beta_k|^2,$$

- Note that the use of l_2 instead of the l_1 norm in the cost function would make the method sensitive to outliers (e.g., impulses).
- Furthermore, the l_1 norm adds some sort of sparsity to the expansion.
- For even more sparse solutions, one may also adopt the l_1 norm for the regularization terms.

Remarks

In the case of the Gaussian Kernel:

$$\|\tilde{f}\|_{\mathcal{H}} = \int_{\mathcal{X}} \sum_{n} \frac{\sigma^{2n}}{n!2^n} (O^n \tilde{f}(x))^2 dx,$$

with $O^{2n} = \Delta^n$ and $O^{2n+1} = \nabla \Delta^n$, Δ being the Laplacian and ∇ the gradient operator.
Thus, we see that the regularization term $\|\tilde{f}\|_{\mathcal{H}}^2$ "penalizes" the derivatives of the minimizer's part that lives on $\mathcal{H}$.

Outline

Selection of the parameters λ, μ

- We keep λ constant.

Selection of the parameters λ, μ

- We keep λ constant.
- The value of μ is adjusted so that:

Selection of the parameters λ, μ

- We keep λ constant.
- The value of μ is adjusted so that:
  - if we are dealing with a pixel-neighborhood that corresponds to a smooth area, μ is large,

Selection of the parameters λ, μ

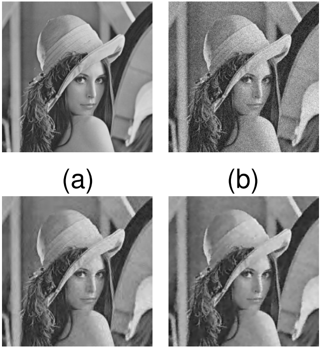
- We keep λ constant.
- The value of μ is adjusted so that:
  - if we are dealing with a pixel-neighborhood that corresponds to a smooth area, μ is large,
  - if we are dealing with a pixel-neighborhood that corresponds to an edge, μ is small,

Selection of the parameters λ, μ

- We keep λ constant.
- The value of μ is adjusted so that:
  - if we are dealing with a pixel-neighborhood that corresponds to a smooth area, μ is large,
  - if we are dealing with a pixel-neighborhood that corresponds to an edge, μ is small,
  - For "steeper" edges, the value of μ is smaller.

Outline

Gaussian Noise Removal



(a) (b)

(c) (d)

Figure: (a) Original Image, (b) Original with additive Gaussian Noise - PSNR=18,7146 dB, (c) Wavelet Denoising (BiShrink) - PSNR=29,3536 dB, (d) Kernelised Denoising - PSNR=29,4535 dB

Impulse Noise Removal

(a) (b)
(c) (d)

Figure: (a) Original Image, (b) Original with additive Impulse Noise - PSNR=12,7562 dB, (c) Wavelet Denoising - PSNR=25,2574 dB, (d) Kernelised Denoising - PSNR=30,1146 dB

Mixed Noise Removal

(a) (b)

Figure: (a) Image with additive mixed Noise (Gaussian + Impulse) - PSNR=21 dB, (b) Kernelised Denoising - PSNR=32,28 dB

Advantages

Extended experiments were conducted using a plethora of cutting edge methods (SKR, BM3D, BiShrink, BLS-GSM, e.t.c.).

Advantages

Extended experiments were conducted using a plethora of cutting edge methods (SKR, BM3D, BiShrink, BLS-GSM, e.t.c.).

Advantages of the kernel based methodology:

Advantages

Extended experiments were conducted using a plethora of cutting edge methods (SKR, BM3D, BiShrink, BLS-GSM, e.t.c.).

Advantages of the kernel based methodology:

- Independence of the noise statistics.

Advantages

Extended experiments were conducted using a plethora of cutting edge methods (SKR, BM3D, BiShrink, BLS-GSM, e.t.c.).

Advantages of the kernel based methodology:

- Independence of the noise statistics.
- Superior results in the presence of impulse or mixed noise.

Advantages

Extended experiments were conducted using a plethora of cutting edge methods (SKR, BM3D, BiShrink, BLS-GSM, e.t.c.).

Advantages of the kernel based methodology:

- Independence of the noise statistics.
- Superior results in the presence of impulse or mixed noise.
- In the presence of gaussian noise, the kernel based method gives results similar to wavelet-based techniques that require no additional information for the noise statistics (such as BiShrink).

Disadvantages

Disadvantages:

Disadvantages

Disadvantages:

- Increased computational complexity.

Disadvantages

Disadvantages:

- Increased computational complexity.
- In the presence of gaussian noise, the cutting edge wavelet-based methods (such as BM3D, BLS-GSM), which require some sort of knowledge of the standard deviation σ, give superior results.

Future Research

- Kernel Based processing in the Wavelet Domain.

Future Research

- Kernel Based processing in the Wavelet Domain.
- Applying the kernel-based approach in the context of super-resolution.