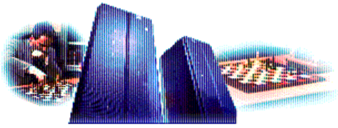
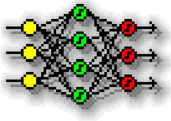




Sistemi Intelligenti
Stima MAP

Alberto Borghese

Università degli Studi di Milano
Laboratory of Applied Intelligent Systems (AIS-Lab)
Dipartimento di Informatica
alberto.borghese@unimi.it

Overview

Statistical filtering

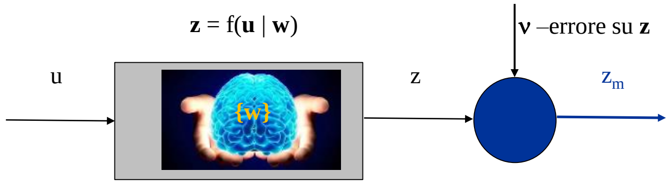
MAP estimate

Different noise models

Different regularizators

Clique

Statistical models



$\nu = z_m - z = z_m - f(u \mid w)$

What happens if u is not exactly known but it is extracted from a statistical distribution?

How can we compute the most likely value of u? And of z_m?

Teorema di Bayes

P(X,Y) = P (Y|X)P(X) = P(X|Y)P(Y)

$$P(X|Y) = \frac{P(Y \mid X)P(X)}{P(Y)}$$

X = causa Y = effetto

$$P(\text{causa}|\text{effetto}) = \frac{P(Effetto| \, Causa)\, P(Causa)}{P(Effetto)}$$

We usually do not know the statistics of the cause, but we can measure the effect and , through frequency, build the statistics of the effect or we know it in advance.

A doctor knows P(Symptons|Causa) and wants to determine P(Causa|Symptoms)

Variabili continue

Caso discreto: prescrizione della probabilità per ognuno dei finiti valori che la variabile X può assumere: p(x).

Caso continuo: i valori che X può assumere sono infiniti. Devo trovare un modo per definirne la probabilità. Descrizione **analitica** mediante la funzione densità di probabilità.

Valgono le stesse relazioni del caso discreto, dove alla somma si sostituisce l'integrale.

$$p(x,y) = p(y|x)\,p(x) = p(x\,|\,y)\,p(y)$$

Teorema di Bayes

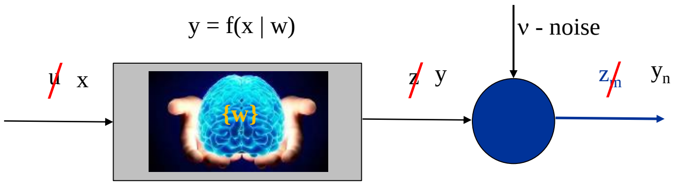
$$p(x\,|\,y) = \frac{p(y|x)\,p(x)}{p(y)}$$

Problema Inverso

x = causa (o parametri del modello)
y = effetto

Obbiettivo

Determinare i dati (la causa, u) più verosimile dato un insieme di misure $z_{m,}$ dato un modello {w}.

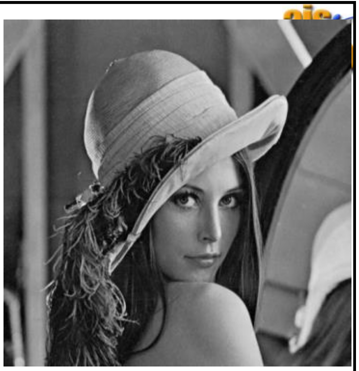


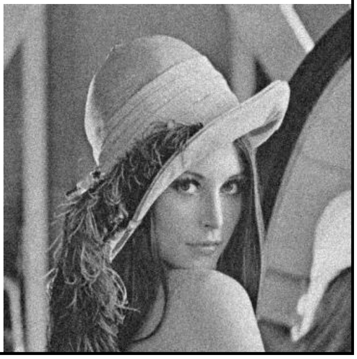
y = Ax se il modello è lineare

Inverse problem: determine cause {x} from {y_n},{w} – utilizzo backwards

Dato y_n posso determinare quale x sia la causa più probabile non solo data la statistica di ν ma anche data la statistica di x.

Images are corrupted by noise…





i) When measurement of some physical parameter is performed, noise corruption cannot be avoided.

ii) Each pixel of a digital image measures a number of photons.

Therefore, from i) and ii)…

…Images are corrupted by noise!

How to go from noisy image, y_n, to the true one, x? It is an inverse problem (true image is the cause, measured image is the measured effect).

Example: Filtering (denoising)

- $x = \{x_1, x_2, \dots, x_M\}$, $x_k \in R^M$ e.g. Pixel true luminance
- $y_n = \{y_{n1}, y_{n2}, \dots, y_{nM}\}$ $y_{nk} \in R^N$ e.g. Pixel measured luminance (noisy)

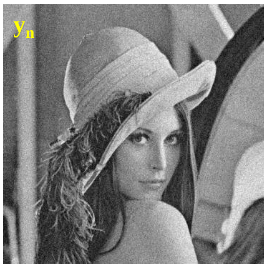
- $y_n = I\ x + n$ ->. Determining x is a **denoising problem** (the measuring device introduces only measurment error) y = Ax => y = Ix.

Role of I:

- Identity matrix. Reproduces the input image, x, in the output y.

Role of n: measurement noise.

- $\mathbf{y_n} = y + n = \mathbf{I}\ \mathbf{x} + n$



Determining x is a denoising problem (image is a copy of the real one with the addition of noise)

Esempio più generale (e.g. deblurring)

- $x = \{x_1, x_2, \ldots, x_M\}$, $x_k \in R^M$ e.g. Pixel true luminance
- $y_n = \{y_{n1}, y_{n2}, \ldots, y_{nM}\}$ $y_{nk} \in R^N$ e.g. Pixel measured luminance (noisy)

- $y_n = y + n = \mathbf{A}\, x + n + h$ -> determining x is a **deblurring problem** (the measuring device introduces easurment error and some blurring)
- **This is the very general equation that describes any sensor.**

Role of A:

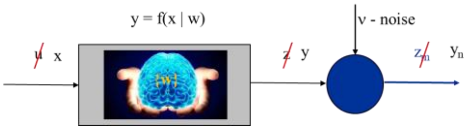
- Matrix that produces the output y_i as a linear combination of other values of x.

Role of h: offset: background radiation (dark currents) has been compensated by calibration, regulation of the zero point.

Role of n: measurement noise.

- $\mathbf{y_n} = y + n = \mathbf{A\, x} + n$ after calibration

Given the measurements $\{y_n\}$ we want to compute the most likelihood set of data $\{x\}$

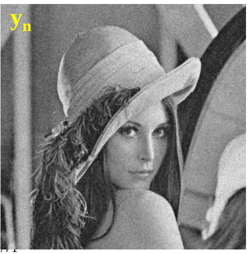


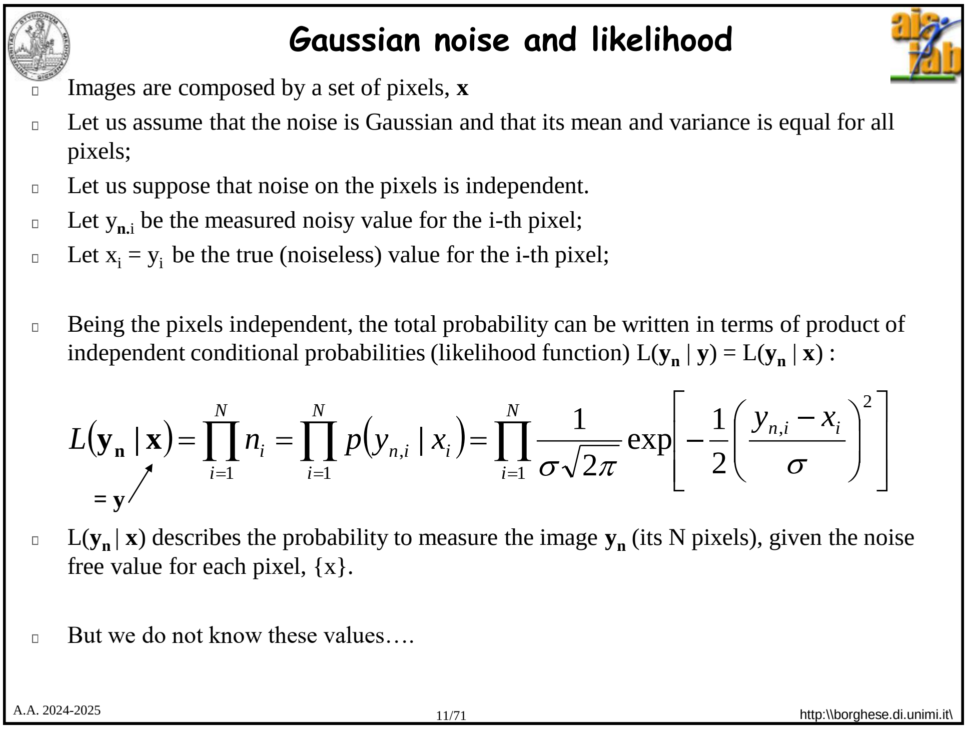
Gaussian noise and likelihood

- Images are composed by a set of pixels, **x**
- Let us assume that the noise is Gaussian and that its mean and variance is equal for all pixels;
- Let us suppose that noise on the pixels is independent.
- Let $y_{\mathbf{n}.i}$ be the measured noisy value for the i-th pixel;
- Let $x_i = y_i$ be the true (noiseless) value for the i-th pixel;

- How can we quantify the probability to measure the image **x**, given the probability density function for the measurement of each pixel $\mathbf{y_n}$?

- Which is the joint probability of measuring the set of pixels: $y_{1n} \ldots y_{Nn}$?



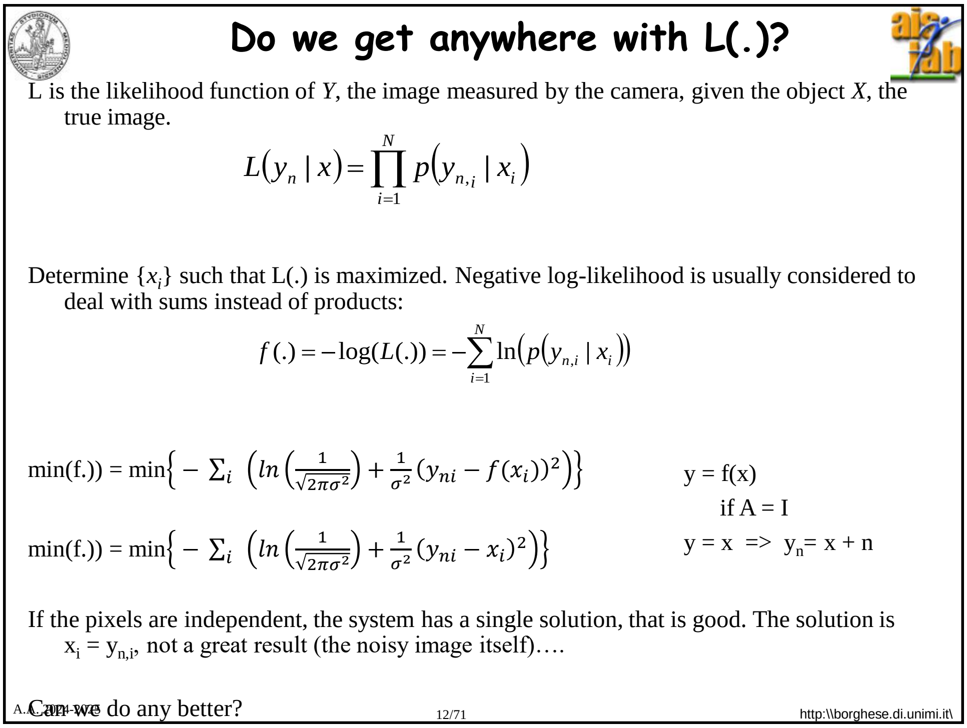


Gaussian noise and likelihood

- Images are composed by a set of pixels, **x**
- Let us assume that the noise is Gaussian and that its mean and variance is equal for all pixels;
- Let us suppose that noise on the pixels is independent.
- Let $y_{\mathbf{n}.\mathrm{i}}$ be the measured noisy value for the i-th pixel;
- Let $x_i = y_i$ be the true (noiseless) value for the i-th pixel;
- Being the pixels independent, the total probability can be written in terms of product of independent conditional probabilities (likelihood function) $L(\mathbf{y_n} \mid \mathbf{y}) = L(\mathbf{y_n} \mid \mathbf{x})$:

$$L(\mathbf{y_n} \mid \underset{= \mathbf{y}}{\mathbf{x}}) = \prod_{i=1}^{N} n_i = \prod_{i=1}^{N} p(y_{n,i} \mid x_i) = \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{y_{n,i} - x_i}{\sigma}\right)^2\right]$$

- $L(\mathbf{y_n} \mid \mathbf{x})$ describes the probability to measure the image $\mathbf{y_n}$ (its N pixels), given the noise free value for each pixel, {x}.
- But we do not know these values….



Do we get anywhere with L(.)?

L is the likelihood function of *Y*, the image measured by the camera, given the object *X*, the true image.

$$L(y_n \mid x) = \prod_{i=1}^{N} p(y_{n,i} \mid x_i)$$

Determine $\{x_i\}$ such that L(.) is maximized. Negative log-likelihood is usually considered to deal with sums instead of products:

$$f(.) = -\log(L(.)) = -\sum_{i=1}^{N} \ln\left(p(y_{n,i} \mid x_i)\right)$$

min(f.)) = min$\left\{-\sum_i \left(ln\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) + \frac{1}{\sigma^2}(y_{ni} - f(x_i))^2\right)\right\}$ y = f(x)

if A = I

min(f.)) = min$\left\{-\sum_i \left(ln\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) + \frac{1}{\sigma^2}(y_{ni} - x_i)^2\right)\right\}$ y = x => y_n= x + n

If the pixels are independent, the system has a single solution, that is good. The solution is $x_i = y_{n,i}$, not a great result (the noisy image itself)….

Can we do any better?

A better approach

$$L(y_n \mid x) = \prod_{i=1}^{N} p\left(y_{n,i} \mid x_i\right)$$

We have N pixels, for each pixel we get **one** measurement.

Let us analyze the probability **for each pixel i indipendently**: $p\left(y_{n,i} \mid x_i\right)$. If we have more measurements for each pixel. **For each pixel,** we can write:

$$p\left(y_{n,i,1}; p_{n,i,2}; p_{n,i,3}; \ldots\ldots p_{n,i,M} \mid x_i\right) = \prod_{k=1}^{M} p\left(y_{n,k,i} \mid x_i\right)$$

Let us analyze the pixel i. If noise is independent, Gaussian, zero mean, the best estimate of x_i is the **samples average**, this converges to the distribution mean of the measurements in the position i.

min(f.)) = $\min\left\{ - \sum_i \left(\frac{1}{\sigma^2} (y_{ni} - x_i)^2 \right) \right\}$

$$x_i = \sum_{k}^{M} y_{n,k,i}$$

From variance analysis, the accuracy of the estimate increases with $\sqrt[2]{M}$ with M number of samples of the same data. The variance decreases with M.

But, **what happens if we do not have such multiple samples** or we have a few samples?

Overview

Statistical filtering

MAP estimate

Different noise models

Different regularizators

Clique

The Bayesian framework

We assume that the object x is a realization of the "abstract" object X that can be characterized statistically as a density probability on X. x is considered extracted randomly from X (a bit Platonic).

The probability $p(y_n| x)$ can be viewed as a conditional probability: $p(y_n| x = x^*)$

That is x will follow also a probability distribution. We will have p(x) = ….

Under this condition, the probability of observing y_n can be written as the joint probability of observing both y_n and x. This is equal to the product of the conditional probability $p(y_n \mid x)$ by a-priori probability on x, p(x):

$$p(y_n, x) = p(y_n \mid x)\, p(x)$$

The Bayesian framework

The probability of observing y_n can be written as the joint probability of observing both y_n and x is equal to the product of the conditional probability $p(y_n \mid x)$ by an a-priori probability on x, p_x:

$$p(y_n, x) = p(y_n \mid x)\, p(x)$$

As we are interested in determining x, **inverse problem**, we have to write the conditional probability of x, having observed (measured) y_n : $p(x \mid y_n)$. We apply Bayes theorem:

$$p(x \mid y_n) = \frac{p(y_n \mid x)\, p(x)}{p(y_n)} = J_0(y_n \mid x) \boxed{\frac{p(x)}{p(y_n)}}$$

where $p(y_n| x)$ is the conditional probability: $J_0 = p(y_n| x = x^*)$

MAP Estimate

Vogliamo trovare il valore dei dati {x} più probabile congruente con le misure {y}:

$$\max_x\{p(x|y_n)\} = \max_x p(y_n|x)\frac{\mathrm{p(x)}}{\mathrm{p}(y_n)}$$

$$\max_x\{\ (\mathrm{x|y_n})\} = \max_x \prod_i p\ (\mathrm{y_{n,i}}\,_{|}\mathrm{x_i})\,\frac{p(\mathrm{x_i})}{p(\mathrm{y_{n,i}})}$$

MAP Estimate

$$p(x\,|\,y_n) = \frac{p(y_n\,|\,x)\,p(x)}{p(y_n)} = L(y_n\,|\,x)\frac{p(x)}{p(y_n)}$$

ln(ab/c) = ln(a) + ln(b) – ln(c)

Logarithms help:

$$-\ln\left(p\left(x\,|\,y_n\right)\right) = -\ln\left\{\frac{\left(p(y_n\,|\,x)\,p(x)\right)}{p(y_n)}\right\} = -\left\{\ln\left(p(y_n\,|\,x)\right) + \ln\left(p(x)\right) - \ln\left(p(y_n)\right)\right\}$$

We maximize the p(x | y_n), by minimizing:

$$\arg\min_x -\left\{\ln\left(\frac{p(y_n\,|\,x)\,p(x)}{p(y_n)}\right)\right\} = \arg\min_x -\left\{\ln\left(p(y_n\,|\,x)\right) + \ln\left(p(x)\right) - \ln\left(\cancel{p(y_n}\right))\right\}$$

We explicitly observe that the marginal distribution of y_n, p(y_n), is not dependent on x. It does not affect the minimization and it can be neglected. It represents the statistical distribution of the measurements alone, implicitly considering all the possible x values.

Maximizing p(x | y_n) is called <u>Maximum A-Posteriori Estimate – MAP</u> (we callect the measurements yn and then we estimate x taking into account also the information on x).

MAP estimate components

We maximize the MAP of p(x | y_n), by minimizing:

$$\arg\min_x -\{\ln(p(y_n \mid x)p(x))\} = \arg\min_x -\{\ln(p(y_n \mid x)) + \ln(p(x))\}$$

$J_0(y_{n,i} \mid x)$

Adherence to the data for each x value (conditional probability)

$J_R(x)$

A-priori probability on x

Depending on the shape of the noise (inside the conditional probability) and the a-priori distribution of x(.), J_R(x), we get **different solutions**.

Gaussian noise on samples

$$x = \arg\min_x -\{\ln(p(y_n \mid x)p(x))\} = \arg\min_x -\{\ln(p(y_n \mid x)) + \ln(p(x))\} =$$

$$\arg\min_x \{J_0(y_n \mid x) + J_R(x)\} =$$

- Gaussian noise on the data
- Zero mean
- All measurements have the same variance, σ_0^2
- Pixels are independent
- y = Ax – deblurring problem (A ≠ I)

$$J_0(y_n \mid x) = \cos\tan te + \left(\frac{1}{\sigma^2}\right)\left(\sum_i \left\| y_{n,i} - Ax_i \right\|^2\right)$$

y_i

-log (p(y_n | x)) = $J_0(y_n|x) = cost + \left(\frac{1}{\sigma^2}\right)\sum_i \|Ax_i - y_{n,i}\|^2$

Mean squared error – errore empirico

What about J_R(x) = -log(p(x))?

Gibb's priors for p(x)

We often define the a-priori term, $J_R(x)$, as Gibb's prior:

$$p_x = \frac{1}{Z}\left\{ e^{\left(-\frac{1}{\beta}U(x)\right)} \right\} \qquad Z = \int_{-\infty}^{+\infty} e^{-\frac{1}{\beta}U(x)}\, dx$$

Integrale = 1

U(x) è solitamente ≥ 0

E' una funzione esponenziale decrescente che è massima quando U(x) è minima
(max $e^{-U(x)}$ si ha quando U(x) = 0)

U(x) sarà perciò minimo per le realizzazioni di x (dell'immagine) più probabili.

U(x) è chiamato anche potenziale => potenziale minimo per realizzazioni più probabili.

Gibb's priors for p(x)

We often define the a-priori term, $J_R(x)$, as Gibb's prior:

$$p_x = \frac{1}{Z}\left\{ e^{\left(-\frac{1}{\beta}U(x)\right)} \right\} \qquad Z = \int_{-\infty}^{+\infty} e^{-\frac{1}{\beta}U(x)}\, dx$$

Considerando il negativo del logaritmo di p(x):

$$J_R(x) = -\ln\left(p_x\right) = +\ln\left(Z\right) + \frac{1}{\beta}U(x)$$

⇒ $J_R(x)$ is a linear function of the potential U(x). It is minimum when U(x) is minimum.

Z does not depend on x => it is constant

β is a constant that provides a scale to $J_R(x)$.

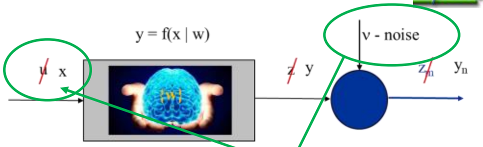
β Explains how p(x) decreases with the decrease of the probability of x, described by U(x)).

A-priori types – p(x)

p(x) describes the probability of having a certain type of data X. In this case it describes the probability of having one image or another.

- It can be the amplitude of the signal defined in terms of power.
- It can be the geometrical structure defined in terms of variations (gradients)
- It can be information gathered from the neighbour data (e.g. clique).
- Any statistical information on the distribution of x.
- It can be a morphable model
- …..

MAP estimate components



We maximize the MAP of p(x | y$_n$), by minimizing:

$$\arg\min_x -\{\ln(p(y_n \mid x)p(x))\} = \arg\min_x -\{\ln(p(y_n \mid x)) + \ln(p(x))\}$$

$J_0(y_{n,i} \mid x)$

Adherence to the data for each x value (conditional probability)

$J_R(x)$

A-priori probability on x

Depending on the shape of the noise (inside the joint probability) and the a-priori distribution of x(.), J$_R$(x), we get different solutions.

P(x) in the Ridge regression

We choose as a-priori term the squared norm of the function x, weighted by P: U(x) = $\|Px^2\|$

$$p(x) = \frac{1}{Z}\left\{ e^{\left(-\frac{1}{\beta}\|Px\|^2\right)} \right\}$$

$J_R(x) = -\log(p(x)) = \log(Z) + (1/\beta)\,\|Px^2\|$

Nel caso del filtraggio: P = I, peso tutti i pixel dell’immagine allo stesso modo (P = I)

$J_R(x) = \log(Z) + (1/\beta)\,\|x^2\|$

Non voglio pixel che “sparino” – non voglio avere dati con valori troppo più elevati degli altri, questi sono improbabili (alto potenziale U(x), basso valore di $J_R(x)$).

La probabilità a-priori è distribuita come una Gaussiana.

Map estimate with U(x) = ||Px||²

$$x = \underset{x}{\operatorname{argmin}}\left(\sum_i \left\|Ax_i - y_{n,i}\right\|^2 + \frac{1}{\beta}\sum_i \left\|p_{ii}x_i \quad\right\|^2\right)$$ Funzione costo quadratica

$J_0\left(y_{n,i} \mid x\right)$

Adherence to the data for each x value (conditional probability)

$J_R\left(x\right)$

A-priori probability on x

MAP estimate with U(x) = ||Px||²

$$x = \underset{x}{\operatorname{argmin}} \left(\sum_i \left\| Ax_i - y_{n,i} \right\|^2 + \frac{1}{\beta} \sum_i \left\| p_{ii} x_i \quad \right\|^2 \right)$$ Funzione costo quadratica

Derivo rispetto a x per calcolare il minimo:

$$x : A^T y_n - A^T A x - \lambda P^T P x = 0 \quad => \quad A^T y_n = \left(A^T A + \lambda P^T P \right) x$$

$J_0\left(y_{n,i} \mid x\right)$ $\qquad$ $J_R\left(x\right)$

Pongo λ = 1/β

Without $\lambda\, P^T P$ large values of x are obtained where $A^T A$ is small. These are reduced by $\lambda\, P^T P$

Map estimate with U(x) = ||Px||²

$$x = \underset{x}{\operatorname{argmin}} \left(\sum_i \left\| Ax_i - y_{n,i} \right\|^2 + \frac{1}{\beta} \sum_i \left\| p_{ii} x_i \quad \right\|^2 \right)$$ Funzione costo quadratica

$$x : A^T y_n - A^T A x - \lambda P^T P x = 0 \quad => \quad A^T y_n = \left(A^T A + \lambda P^T P \right) x$$

$\mathbf{x = (A^T A + \lambda P^T P)^{-1} A^T y_n}$ ---

(diventa risolubile anche quando A è singolare! – norma minima della soluzione)
(ottengo una soluzione che «scoraggia» i valori elevati di x).
(per λ = 0 ritorno alla soluzione con la pseudo-inversa, massima verosimiglianza;
non tengo conto del termine a-priori).

Approccio algebrico

A x = b + N $$\sum_k {\nu_k}^2 = \|Ax - b\|^2$$

$$x = \underset{x}{argmin}\left(\sum_i \left\|A_{*,i}x_i - y_{n,i}\right\|^2\right) \Longrightarrow x = (A^TA)^{-1}A^Ty_n$$

Se la matrice di covarianza ha determinante vicino a zero (è mal condizionata) la soluzione può variare molto con il variare dei dati. **Otteniamo un modello la cui validità al di fuori dei dati già disponibili è molto limitata sui dati che si potranno collezionare in futuro.**

Problema mal posto (Hadamard).

- Esiste una soluzione
- La soluzione è unica
- Varia con continuità con i dati (se A^TA è vicino alla singolarità, il problema è mal posto)

Come possiamo stabilizzare la soluzione?

Approccio algebrico: regolarizzazione

A x = b + N $$\sum_k {\nu_k}^2 = \|Ax - b\|^2$$

$$x = \underset{x}{argmin}\left(\sum_i \left\|A_{*,i}x_i - y_{n,i}\right\|^2\right) \Longrightarrow x = (A^TA)^{-1}A^Ty_n$$

We add a penalty term to the solution that expresses the desired characteristics of the solution.

$$x = \underset{x}{\operatorname{argmin}}\left(\sum_i \left\|Ax_i - y_{n,i}\right\|^2 + \lambda \sum_i \left\|Px_i \quad\right\|^2\right)$$

This is the Tikhonov regularization (1963) or ridge regression.

It is the same cost function obtained when maximizing the MAP with Gibbs prior and quadratic potential function.

Which is the most adequate p(x) for images?

We are very interested to borders, structure. This has to deal with **gradients**.
=> we look at **differential properties**.

We look at the local gradient of the image: ∇x (variazioni spaziali).

One possibility is to use the square of the gradient as a regularizer (o come funzione potenziale): $|| \nabla x ||^2$

This is another form of Tikhonov regularization.

Differential Gibbs prior

$$p_x = \frac{1}{Z}\left\{ e^{\left(-\frac{1}{\beta}U(x)\right)} \right\} \qquad Z = \int_{-\infty}^{+\infty} e^{-\frac{1}{\beta}U(x)} dx$$

$$U(x) = \; || \nabla x ||^2$$

$$\arg\min_x \left\{ \left\| (Ax - y_n)^2 \right\| + \lambda \left\| \nabla x \right\|^2 \right\}$$

$$x: \; \left\{ 2A^T \left(Ax - y_n \right) + 2\lambda \nabla x \right\} = 0$$

System of M linear differential equations. How does it become in the discrete case?

Differential Gibbs prior

$$\arg\min_{x} \left\{ \left\| \left(Ax - y_n\right)^2 \right\| + \lambda \left\| \nabla x \right\|^2 \right\}$$

$$x: \ \left\{ 2A^T \left(Ax - y_n\right) + 2\lambda \nabla x \right\} = 0$$

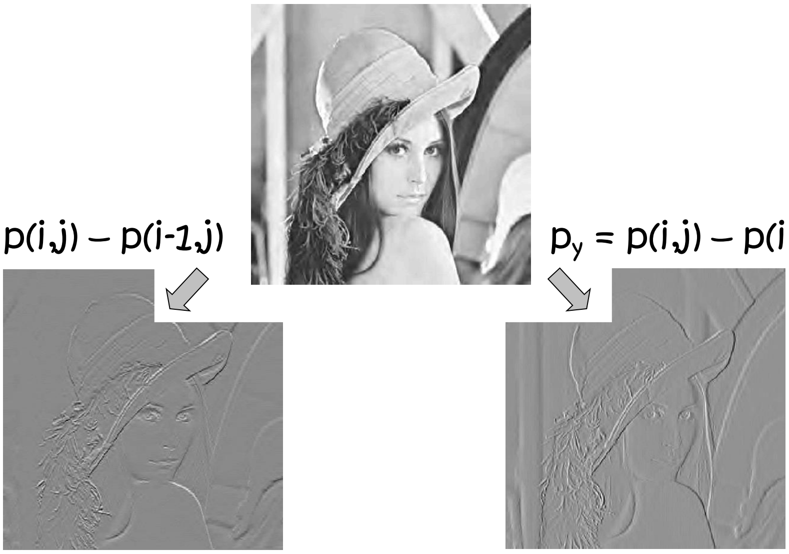
If we apporximate ∇x with the finite differences, one possibility is the following:

$$\| \nabla x_{i,j} \|^2 = \left(x_{i+1,j} - x_{i-1,j}\right)^2 + \left(x_{i,j+1} - x_{i,j-1}\right)^2$$ Centered discrete gradient

$$\arg\min_{x} \left\{ \sum_j \sum_i \left(A_{ji} x_i - y_j \right)^2 + \lambda \left((x_{i,j+1} - x_{i,j-1})^2 + (x_{i+1,j} - x_{i-1,j})^2 \right) \right\}$$

Si può calcolare la derivate della somma, derivando per ciascun elemento x e ponendo la derivate uguale a zero. Diventa un sistema lineare.

A priori term – image gradients (no noise)



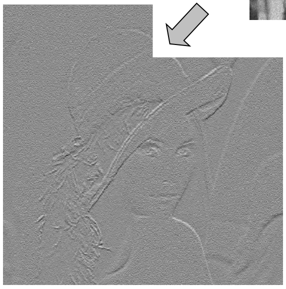
$p_x = p(i,j) - p(i-1,j)$

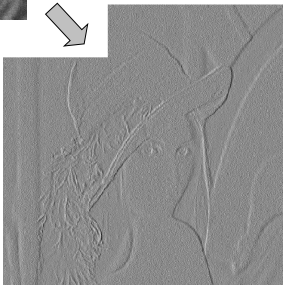
$p_y = p(i,j) - p(i,j-1)$

A priori term – image gradients (with noise)

$$\Delta x_{row} = \frac{x_{i+1,j} - x_{i-1,j}}{2}$$

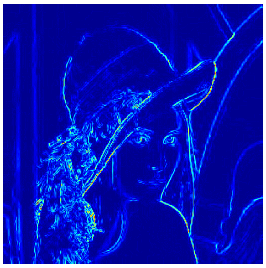
$$\Delta x_{col} = \frac{x_{i,j+1} - x_{i,j-1}}{2}$$



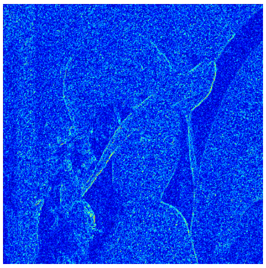


A priori term – norm of image gradient

No noise



Noise



In the real image, most of the areas are characterized by an (almost) null gradient norm. When noise is added, local gradients appear everywhere in the image (real case).

We can for instance suppose that the noise is a random variable with Gaussian distribution, zero mean and variance equal to β^2 (sampling noise).

Overview

Statistical filtering

MAP estimate

Different noise models

Different regularizators

Clique

MAP estimate components

We maximize the MAP of p(x | y$_n$), by minimizing:

$$\arg\min_x -\{\ln(p(y_n \mid x)p(x))\} = \arg\min_x -\{\ln(\underbrace{p(y_n \mid x)}_{J_0(y_{n,i} \mid x)}) + \ln(\underbrace{p(x)}_{J_R(x)})\}$$

Adherence to the data for each x value (conditional probability)

A-priori probability on x

Tikhonov regularization

$$x = \underset{x}{\operatorname{argmin}} \left(\sum_i \left\| Ax_i - y_{n,i} \right\|^2 + \lambda \sum_i \left\| Px_i \quad \right\|^2 \right)$$

Ridge regression

$$x = \underset{x}{\operatorname{argmin}} \left(\sum_i \left\| Ax_i - y_{n,i} \right\|^2 + \lambda \sum_i \left\| \nabla x_i \quad \right\|^2 \right)$$

It is a quadratic cost function. We find *x* minimizing with respect to x the cost function.

This approach is derived in the domain of mathematics. It leads to the same cost function of the MAP approach.

Departing from Tikhonov regularization: different noise models

$$\underset{x}{\arg\min} \; -\{\ln(p(y_n \mid x)\,p(x))\} = \underset{x}{\arg\min} \; -\{\underbrace{\ln(p(y_n \mid x))}_{J_0(y_{n,i} \mid x)} + \underbrace{\ln(p(x))}_{J_R(x)}\}$$

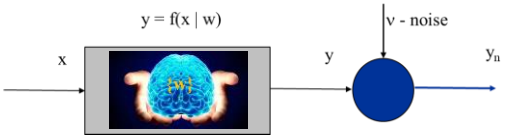
Adherence to the data for each x value (conditional probability)

Two actors:

- $J_0(y_{n,i} \mid x)$ Conditional probability of having the measurements given a certain input.
  - **We can have different noise models.**

$J_R(x)$ - Probability of having a certain solution.
  - **We can have different regularizers**

Different noise models



y = f(x) => y = Ax

Gaussian noise:

Square regularization

Tikhonov $J_0(y_{n,i} \mid x) = \|Ax - b\|^2$ $J_R(x)$ = (1/β) ||Px²||

Ridge regression $J_0(y_{n,i} \mid x) = \|Ax - b\|^2$ $J_R(x)$ = (1/β) ||x²||

$J_0(y_{n,i} \mid x) = \sum_i \|Ax_i - y_{n,i}\|^2$ Empirical error

Kullback-Leibler divergence

Poisson noise:

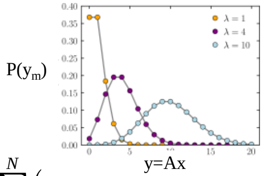
$$J_0(y_{n,i} \mid x) = \sum_i y_{n,i} \ln\left(\frac{y_{n,i}}{Ax} + Ax_i - y_{n,i}\right)$$

KL and the Poisson noise

$\nu_i = \|A\ x - y_{ni}\ \|$

We know the statistical distribution of the noise inside the conditional probability of y_{ni} given x.

For one pixel: $p(y_{ni} \mid x_i) = \left\{ \frac{e^{-Ax_i} (Ax_i)^{y_{n_i}}}{y_{n_i}!} \right\}$



$$-\ln(L(y_n; x)) = -\ln\left(\prod_{i=1}^{N} p(y_{n,i}; x_i)\right) = -\sum_{i=1}^{N}\left(-Ax_i + y_{n,i}\ln(Ax_i) - \ln(y_{n,i}!)\right)$$

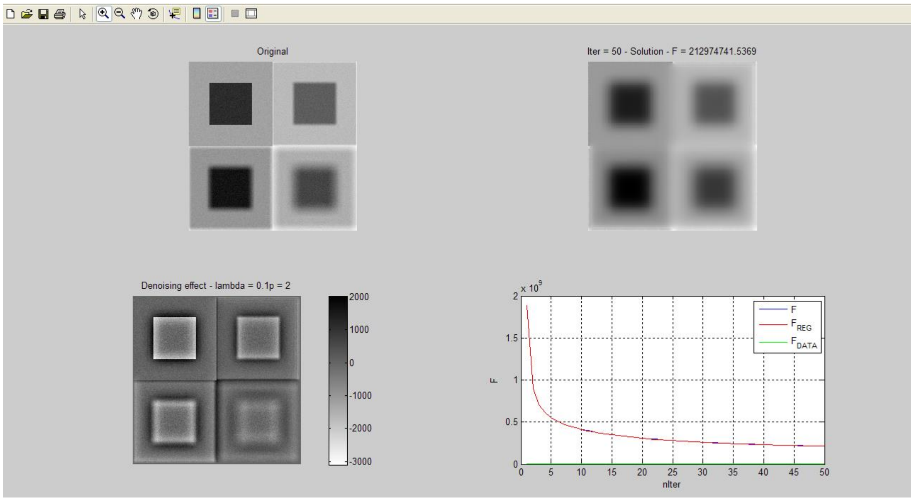
To eliminate the factorial term, we normalize the likelihood by $L(y_n, y_n)$:

$$-\ln\left(\frac{L(y_n, x)}{L(y_{n,} y_n)}\right) = -\sum_{i=1}^{N}\left(y_n \ln(Ax) - \ln(y_n) + y_n - Ax\right) = KL\ divergence$$

$$= \sum_i y_n \ln\left(\frac{y_n}{Ax} + Ax - y_n\right)$$

It is not a distance!
It is not linear

Tikhonov regularization - simulations

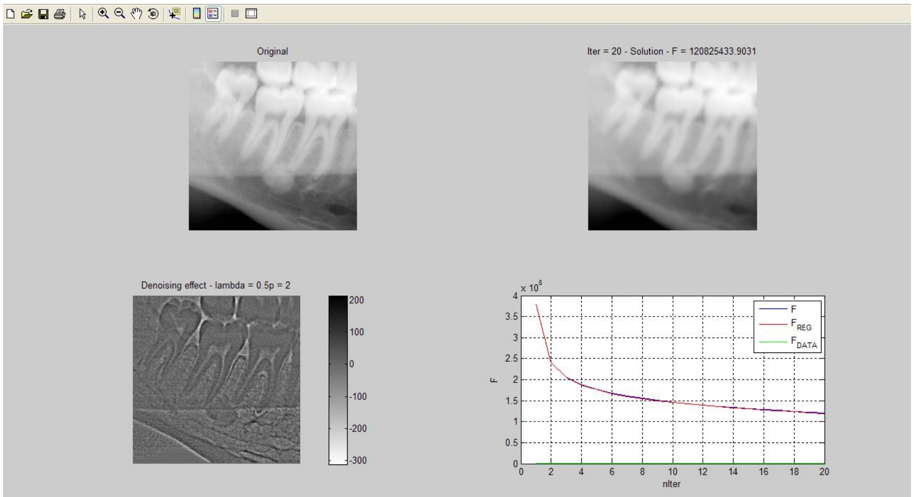


Edge smoothing effect with Tikhonov-like regularizator (P(x) = || ∇x ||^2)

Poisson noise on the image – λ = 0.5. KL is applied in the first term.

P is the gradient operator

Tikhonov regularization – panoramic images

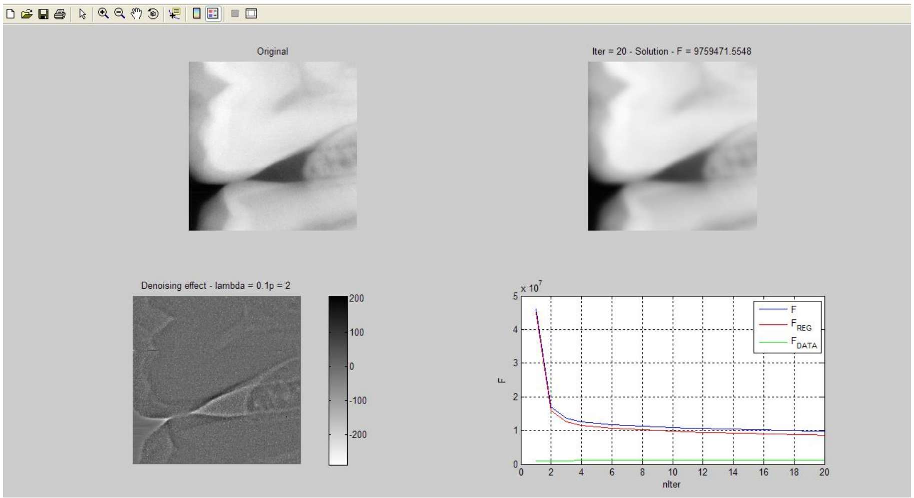


Edge smoothing effect with Tikhonov-like regularization

Poisson noise model - λ = 0.5. KL is applied in the first term.

P is the gradient operator

Tikhonov regularization - endo-oral images


Original
Iter = 20 - Solution - F = 9759471.5548
Denoising effect - lambda = 0.1p = 2
200
100
0
-100
-200
x 10^7
5
4
3
2
1
0
F
0
2
4
6
8
10
12
14
16
18
20
nIter
F
F_REG
F_DATA

Edge smoothing effect with Tikhonov-like regularization

Poisson noise model - $\lambda = 0.1$ (less regularization)

KL is applied in the first term.

P is the gradient operator

Overview

Statistical filtering

MAP estimate

Different noise models

Different regularizators

Clique

Departing from Tikhonov regularization: different regularizers

$$\arg\min_x -\{\ln(p(y_n \mid x)\,p(x))\} = \arg\min_x -\{\ln(p(y_n \mid x)) + \ln(p(x))\}$$

$J_0(y_{n,i} \mid x)$

Adherence to the data for each x value (conditional probability)

$J_R(x)$

Two actors:

- $J_0(y_{n,i} \mid x)$ Conditional probability of having the measurements given a certain input.
  - **We can have different noise models.**

$J_R(x)$ - Probability of having a certain solution.
  - **We can have different regularizers**

Non-quadratic a-priori: norm l_2

$$\arg\min_x -\{\ln(p(y_n \mid x)\,p(x))\} = \arg\min_x -\{\ln(p(y_n \mid x)) + \ln(p(x))\}$$

$J_0(y_{n,i} \mid x)$

Adherence to the data for each x value (conditional probability)

$J_R(x)$

$$J_R(x) \quad = \quad \sum_i \sqrt[2]{x_1^2 + x_2^2 + \dots x_N^2}$$

Norma l_2 di x
La norma l_2 di **x** è minima.

Non-quadratic a-priori: total variation

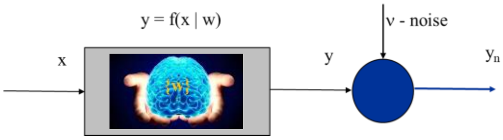
$$\arg\min_x -\{\ln(p(y_n \mid x)\,p(x))\} = \arg\min_x -\{\underbrace{\ln(p(y_n \mid x))}_{J_0(y_{n,i} \mid x)} + \underbrace{\ln(p(x))}_{J_R(x)}\}$$

Adherence to the data for each x value (conditional probability)

$$J_R(x) \;=\; \sqrt[2]{\Delta x_1^2 + \Delta x_2^2 + \dots \Delta x_N^2}$$

Norma l_2 delle variazione di x o variazione totale di x (**total variation**)
Il modulo della somma degli incrementi (in valore assoluto) di x è minima.

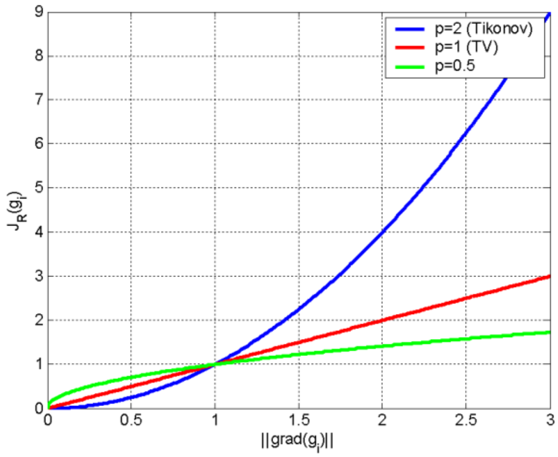
Different a-priori



y = f(x) => y = Ax

	Noise model	Gaussian	Regularizer
Tikhonov	$J_0(y_{n,i} \mid x)$	$= \|\|Ax - b\|\|^2$	$J_R(x) = (1/\beta)\ \|\|Px^2\|\|$
Ridge regression	$J_0(y_{n,i} \mid x)$	$= \|\|Ax - b\|\|^2$	$J_R(x) = (1/\beta)\ \|\|x^2\|\|$
l_2 (total variation) regularization	$J_0(y_{n,i} \mid x)$	$= \|\|Ax - b\|\|^2$	$J_R(x) = (1/\beta)\sqrt[2]{\Delta x_1^2 + \Delta x_2^2 + \dots \Delta x_N^2}$
Lasso regression	$J_0(y_{n,i} \mid x)$	$= \|\|Ax - b\|\|^2$	$J_R(x) = (1/\beta)\left(\|\Delta x_1\| + \|\Delta x_2\| + \cdots + \|\Delta x_N\|\right)$

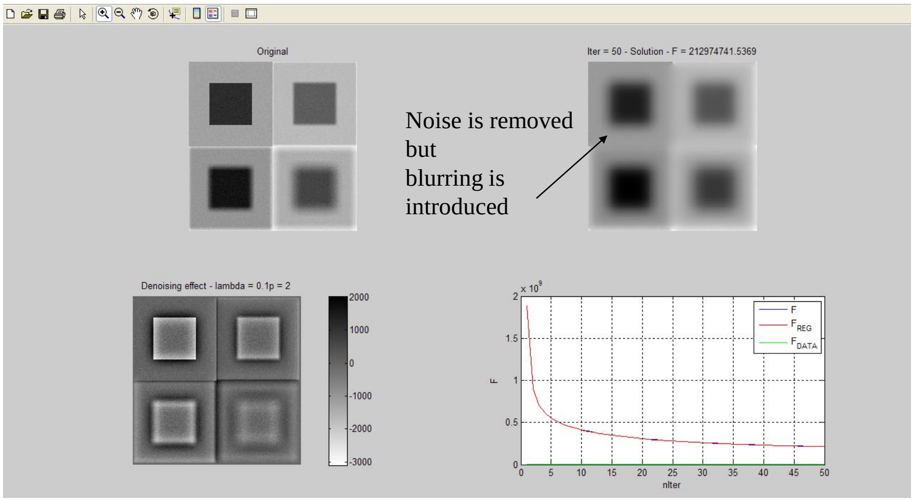
Cost introduced by the regularization term



Cost increases quadratically with the local gradient in Tikhonov
Cost increases linearly with the local gradient in Total Variation (TV)

For this reason TV regularizer is considered "edge preserving" (structure preserving)

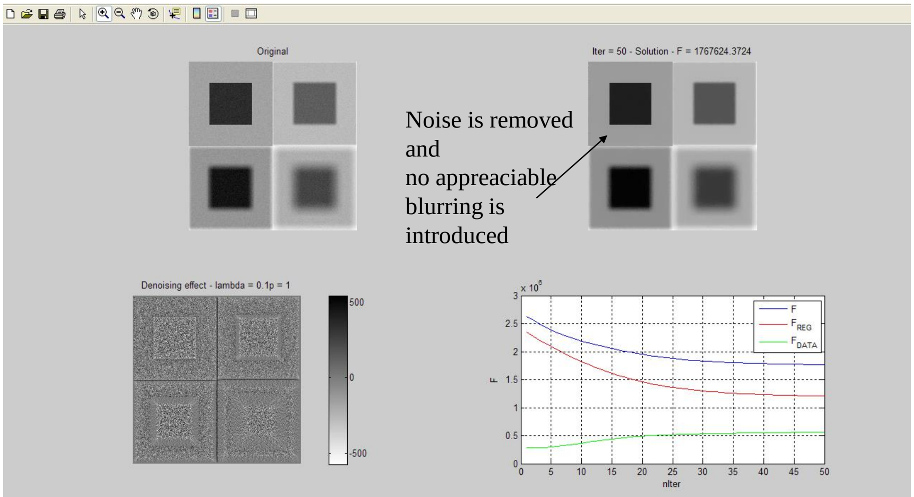
Tikhonov regularization - simulations



Noise is removed
but
blurring is
introduced

Edge smoothing effect with Tikhonov-like regularization
Poisson noise model – $\lambda = 0.5$
P is the gradient operator

Total variation regularization - simulations


Original
Iter = 50 - Solution - F = 1767624.3724
Denoising effect - lambda = 0.1p = 1
500
0
-500
x 10^6
3
2.5
2
1.5
1
0.5
0
F
0
5
10
15
20
25
30
35
40
45
50
nIter
F
F_REG
F_DATA

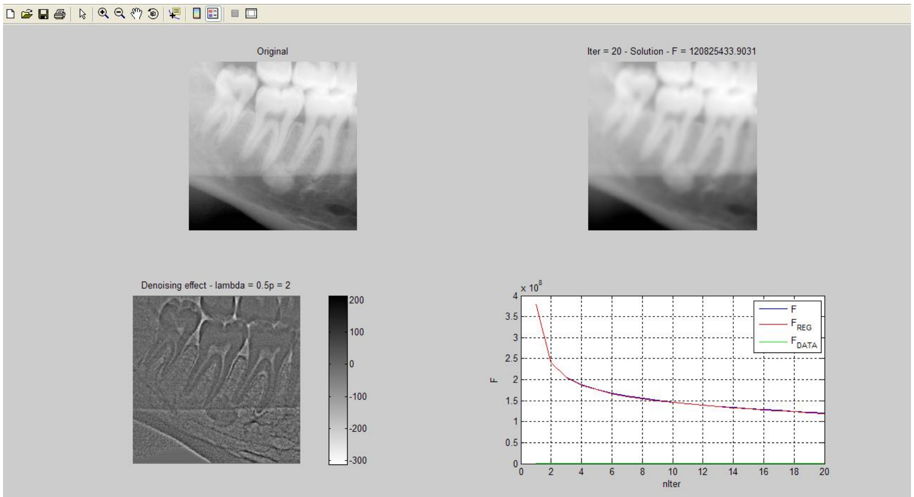
Noise is removed
and
no appreaciable
blurring is
introduced

No appreciable edge smoothing with total variation regularizer

Poisson noise model - $\lambda = 0.5$

P is the gradient operator

Tikhonov regularization – panoramic images

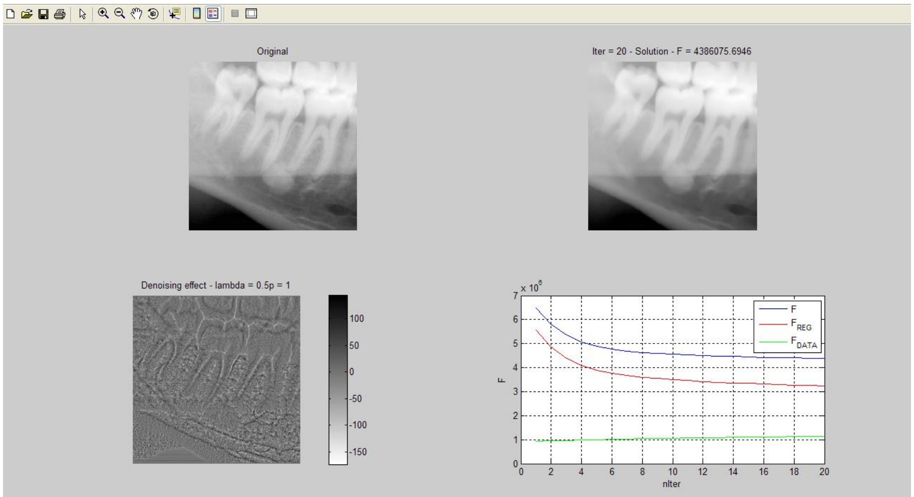


Edge smoothing effect with Tikhonov-like regularization

Poisson noise model - $\lambda = 0.5$

P is the gradient operator

Total variation regularization – panoramic images

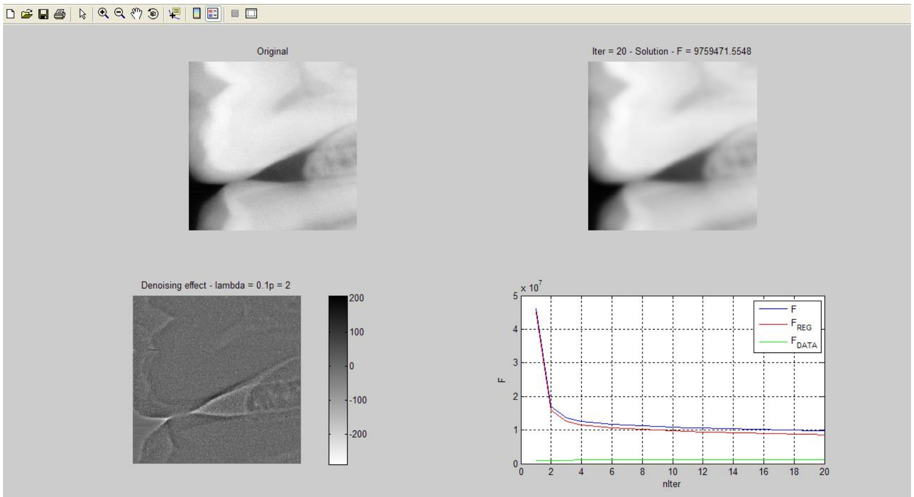


No appreciable edge smoothing with total variation regularizer

Poisson noise model - $\lambda = 0.5$

P is the gradient operator

Tikhonov regularization - endo-oral images

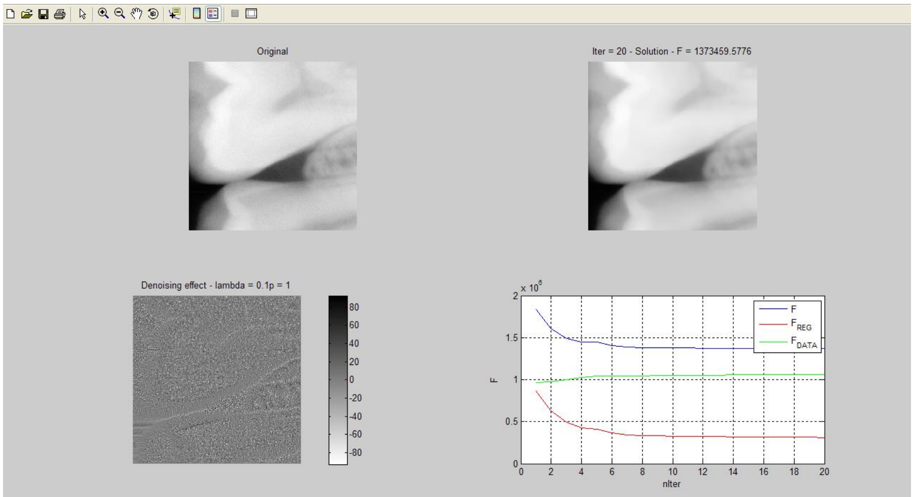


Edge smoothing effect with Tikhonov-like regularization

Poisson noise model - $\lambda = 0.1$

P is the gradient operator

Total variation – endo-oral images



No appreciable edge smoothing with total variation regularizer

Poisson noise model - $\lambda = 0.1$

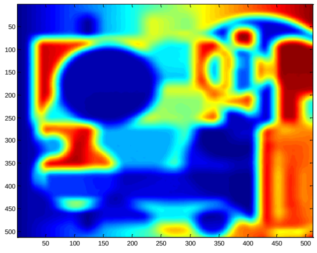
P is the gradient operator

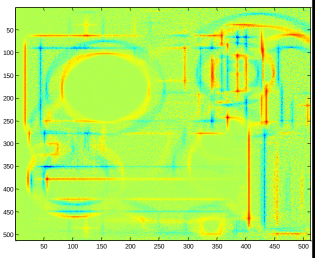
Tikhonov vs. TV (preview)

Filtered image

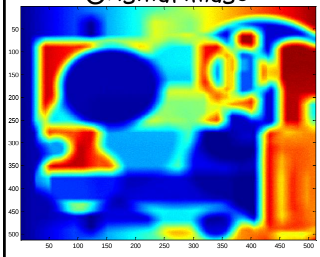
Difference

Tikhonov =>

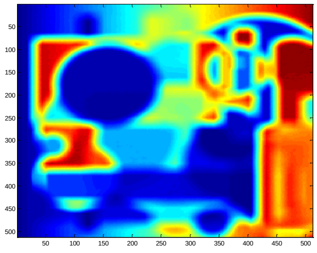


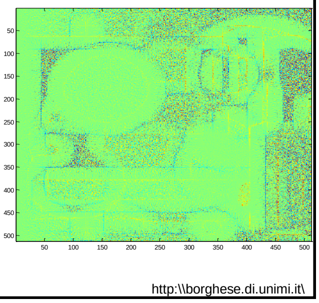


Original image



TV =>





Open problems in TV

- Better images with TV regularizer, but:

Non linear cost functions (non quadratic) also with Gaussian noise model

$$x = \arg\min_x \sum_i \left(\|y_n - Ax\|^2 + \lambda \sqrt{\sum_p^P x_{p,i}{}^2} \right)$$

Minimization does not lead to a linear function (because of the square root) ➔ It requires non-linear iterative minimization.

The derivative of a square root provides a function of the type k/sqrt(.)

Singularity in x = 0 → x ≠ 0

We can use algorithms for constrained minimization (solution should stay inside the first quadrant, e.g. split gradient).

How to set the regularization parameter (λ = 1/β)

$$\boxed{J(f) = J_o(f) + \lambda J_R(f)}$$

$$\arg\min_x -\{\ln(p(y_n \mid x)p(x))\} = \arg\min_x -\{\ln(p(y_n \mid x)) + \ln(p(x))\}$$

	Noise model	Gaussian	Regularizer
Tikhonov	$J_0(y_{n,i} \mid x)$	$= \|\|Ax - b\|\|^2$	$J_R(x)$ = (1/β) ||Px²||
Ridge regression	$J_0(y_{n,i} \mid x)$	$= \|\|Ax - b\|\|^2$	$J_R(x)$ = (1/β) ||x²||
l_2 (total variation) regularization	$J_0(y_{n,i} \mid x)$	$= \|\|Ax - b\|\|^2$	$J_R(x) = (1/\beta) \sqrt[2]{\Delta x_1^2 + \Delta x_2^2 + \dots \Delta x_N^2}$
Lasso regression	$J_0(y_{n,i} \mid x)$	$= \|\|Ax - b\|\|^2$	$J_R(x) = (1/\beta) \left(\|\Delta x_1\| + \|\Delta x_2\| + \cdots + \|\Delta x_N\| \right)$

Role of λ

$$K(\sigma)\sum_i \left\| g_{n,i} - Af_i \right\|^2 \qquad\qquad -\ln\left\{ \frac{1}{Z} e^{\left\{-\frac{1}{\beta}U(\mathbf{f})\right\}} \right\}$$

$$\boxed{J(x) = J_0(x) + \lambda J_R(x)}$$

λ incorporates different elements here:
- the standard deviation of the noise in the likelihood
- the "temperature", that is the decrease in the energy of the configurations with their cost (β)
- the normalized constant Z.

λ has been investigated in the classical regularization theory (Engl et al., 1996), but not as deep in the Bayesian framework ➔ λ is set experimentally through cross-validation.

How to set the regularization parameter – Gaussian case

Analysis of the residual after the estimate **n** = **y- Ax**

- The residual should be distributed as the noise distribution

Gaussian case:
Start with λ = 0 -> x minimizzerà la likelihood $J_0(x) = 0$ (n = 0).

Is this a good solution? No!!

$$J(x) = ||Ax - b||^2 + \lambda \sqrt[2]{\Delta x_1^2 + \Delta x_2^2 + \dots \Delta x_N^2}$$

$$J(x) = J_0(x) + \lambda J_R(x)$$

We are reconstructing the data **and** the error. The latter is usually rapidly varying (e.g. grain images)

We get a better result if we throw away from x the error. This happens when n ≠ 0. Increasing λ, we penalize rapid variations -> $J_0(x)$ increases, n increases -> it approaches the shape of the measurement error.

We stop when
- $(r_i, r_j) = \Sigma^2$ $(||r||^2 = \sigma^2)$
- Sample covariance is equal to distribution covariance
- Average value of the residual is zero,

How to set the regularization parameter – Poisson case

Analysis of the residual after the estimate **n** = **y - Ax**

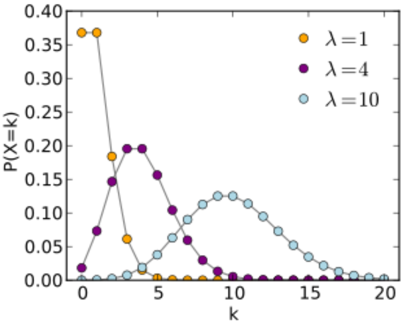
- The residual should be distrubuted as the noise distribution

Poisson case:

- r_i tends to be larger, the larger is x_i.
- λ is increased until $||r||^2 / \mu$ -> 1 (the mean is equal to variance)

1 parametro (media = varianza):

$\mu = \sigma^2$



Overview

Statistical filtering

MAP estimate

Different noise models

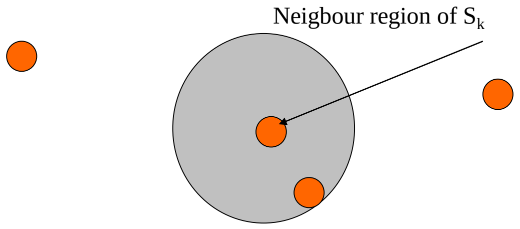
Different regularizators

Clique

A-priori on cliques

We can insert in the a-priori term all the desirable characteristic of the image: local smoothness, edges, piece-wise constancy,….

The idea of defining a **neighboring system** is a natural one:



Images have a natural neighbouring system: the pixels structure. We want to consider the local properties of the image considering neighboring pixels (in particular differential properties - our vision system is particularly tuning to gradients both spatial and temporal). Ideas have been borrowed from physics.

Neighboring System

Let P be the set of pixels of the image: $P = \{p_1, p_2, \ldots p_P\}$

The neighboring system defined over P, S, is defined as $H = \{\mathcal{N}_p \mid p, \forall p \in P\}$, that has the following properties:

An element is not a neighbour of itself: $p_k \notin \mathcal{N}_{pk}$

Mutuality of the neighboring relationship: $p_k \in \mathcal{N}_{pj} \leftarrow\rightarrow p_j \in \mathcal{N}_{pk}$

(S, P) constitute a graph where P contains the nodes of the graph and S the the links.

Depending on the distance from p, different neighboring systems can be defined:

	o	
o	x	o
	o	

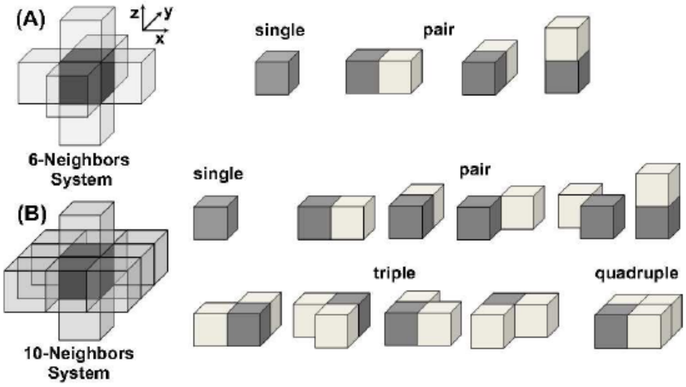
First order neighboring System
4-neighboring System

o	o	o
o	x	o
o	o	o

Second order neighboring System
8-neighboring System

Clique

Borrowed from phisics.



A clique C, for (S, P), is defined as a subset of sites in S.
I can considered ordered sets of voxels, that are connected to p through S.
Types of cliques: single-site, pairs of neighboring sites, triples of neighboring sites,… up to the cardinality of $\mathcal{N}_p$

Markov Random Field

Given (S, P) we can define a set of random values, $\{f_k(p)\}$ for each element defined by S, that is in $\mathcal{N}_p$. Therefore we define a **random field** , $\mathcal{F}$, over S:

$\mathcal{F}(\mathcal{N}_p) = \{f_k(m) \mid m \in \mathcal{N}_p \} \; \forall p$

Under the Markovian hypotheses:

$P(f(p)) \geq 0 \; \forall p$	Positivity
$P(f(p) \mid g(P\text{-}\{p\}) = P(f(p) \mid g(\mathcal{N}_p)\}$	Markovianity

2 expresses the fact that the probability of p assuming a certain value, f (e.g. a certain gradient), is the same considering all the pixel of P but p or only the neighbor pixels is the same, that is the value of f depends only on the gray value of the pixels in $\mathcal{N}_{p.}$

the random field $\mathcal{F}$ is named **Markov Random Field**.

Energy in a Markov Random Field

A “potential” function, $\phi(f)$, can be defined for a MRF. This is a scalar value that is a function of the random value associated to the pixels for all the possible elements of a clique:

$$\phi_c(f) = \sum_{j \in c} f(p_j)$$

If we consider all the possible cliques defined for each element p, we can define a potential energy function associated to the MRF:

$$U(f) = \sum_{c \in C} \phi_c(f)$$

The higher is the potential energy, the lower is the probability that the set of random values of the elements of the cliques is realized, that is the higher is the penalization for the associated configuration.

Gibbs prior

If we consider all the possible cliques defined for each element p, we can define a potential energy function associated to the MRF:

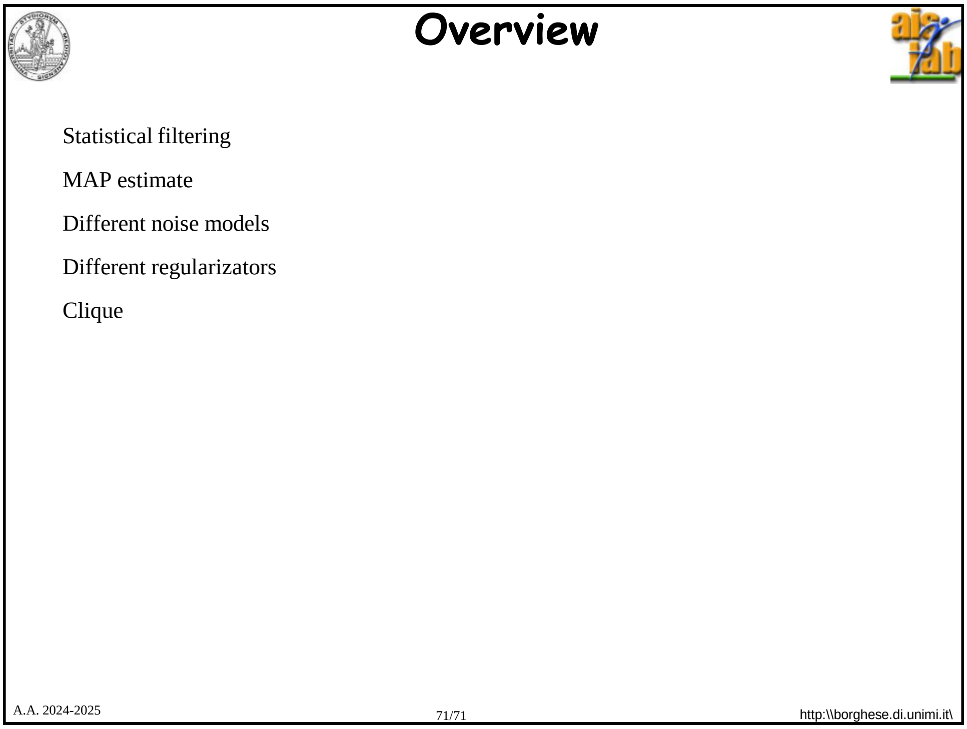
$$U(\mathbf{f}) = \sum_{c \in C} \phi_c(\mathbf{f})$$

The higher is the potential energy, the lower is the probability that the set of random values of the elements of the cliques is realized, that is the higher is the penalization for the associated configuration.

This is well captured by the Gibbs distribution, that describes the probability of a certain configuration to occur. It is a function exponentially decreasing of U:

$$P(\mathbf{f}) = \frac{1}{Z} e^{\left\{-\frac{1}{\beta} U(\mathbf{f})\right\}}$$

P(f) is a Gibbs random field, Hammersley-Clifford theorem (1971). β regulates the decrease in probability and it is associated with temperature in physics. Z is a normalization constant. NB to define Gibbs random fields, $P(f) > 0$, $P(f) \rightarrow 0$ $U(f) \rightarrow \infty$: there are not configurations with 0 probability.



Overview

Statistical filtering

MAP estimate

Different noise models

Different regularizators

Clique