ON "GLOBALLY-OPTIMAL" LOCAL MODELING

from Moving Least Squares to Overparameterization

Alfred M. Bruckstein

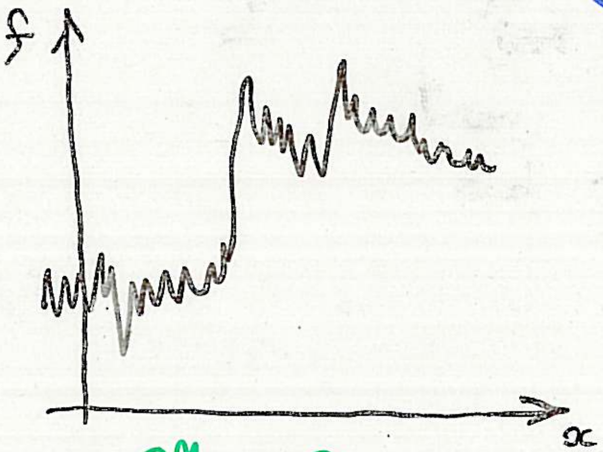
NTU / Technion
Singapore / Israel

The Local Modeling Idea

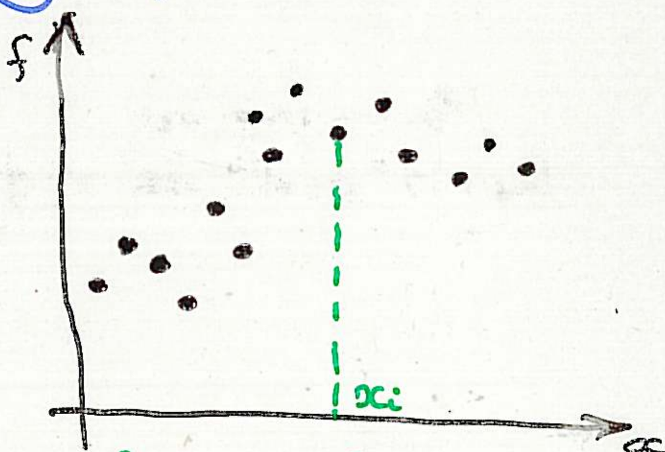
<u>Given</u> data:

A noisy signal

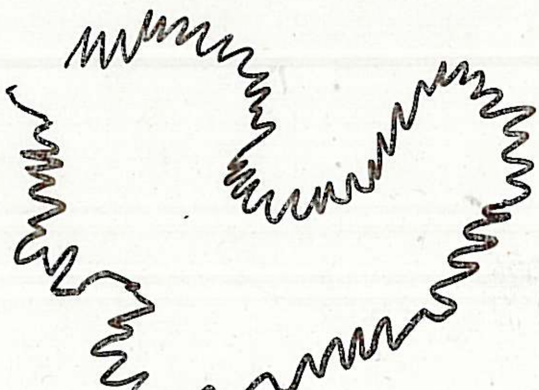
Scattered noisy samples of signal



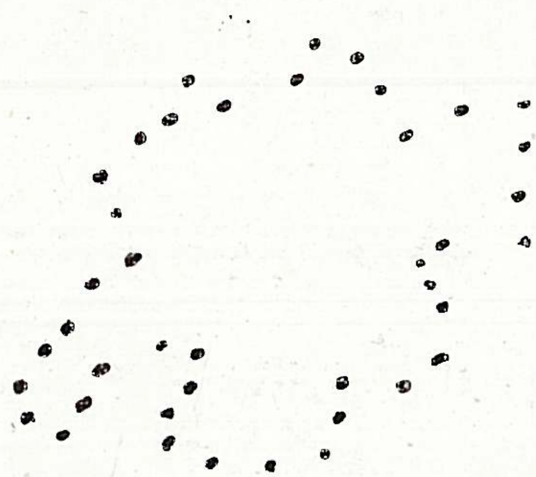
$f^n(x) = f(x) + n(x)$



$\{(x_i, f_i = f(x_i) + n_i)\}$



A noisy curve



Noisy sample of a curve

Local Modeling

In some "neighborhood" the data is a noisy signal which is well described by a (local) parameterized model of the form

$$\text{Signal} = \sum_{i=0}^{K} a_i \phi_i$$

where a_i's are parameters and ϕ_i's are "basis" signals

The term "neighborhood" will be here interpreted: the part of the signal that is near a certain location in the space where the signal lives!

Then local modeling will be a process of estimating a_i's from the noisy signal or its samples.

The "royal" example: local linear or polynomial approximation

$$f(x) \cong \underbrace{\underbrace{f(x_0) + \frac{1}{1!} f'(x_0)(x-x_0)}_{m(x_0)\,x + n(x_0)} + \frac{1}{2!} f''(x_0)(x-x_0)^2}_{a(x_0)\,x^2 + b(x_0)\,x + c(x_0)} + \ldots$$

Note that we have

$$f(x) = \sum_{i=0}^{k} a_i \phi_i(x)$$

but better expressed

$$f(x; x_0) = \sum_{i=0}^{k} a_i(x_0)\, \phi_i(x)$$

and

$$f(x_0; x_0) = \sum_{i=0}^{k} a_i(x_0)\, \underbrace{\phi_i(x_0)}_{\text{the "basis" functions } \{1, x, x^2, \ldots\}}$$

Suppose we want to use this local modeling for smoothing / denoising / analysing the noisy data $f^n(x)$ or $\{(x_i, f_i)\}$

Then one can do

at $x = x_0$

estimate $a_i(x_0)$ in the "local" model

$$f(x) = \sum_{i=0}^{k} a_i \phi_i(x)$$

by minimization:

$$\hat{\Theta}[=a_0 a_1 .. a_k] = \underset{[a_0 a_1 ... a_k]}{\arg\min} \text{ Distance}\left[f^n, f = \sum_{i=0}^{k} a \phi_i\right]$$

$$\left[\text{or Distance}\left[\{f_i, f(x_i) = \sum_{i=0}^{k} a_i \phi_i(x_i)\right]\right]$$

Then the best local estimate (at x_0) of $f(x)$ will be

$$\hat{f}(x_0) = \sum_{i=0}^{k} a_i(x_0) \phi_i(x_0)$$

now do this for every x_0 to get a "MOVING" best estimate of f.

In the above process the distance between the data and the model should be carefully selected:

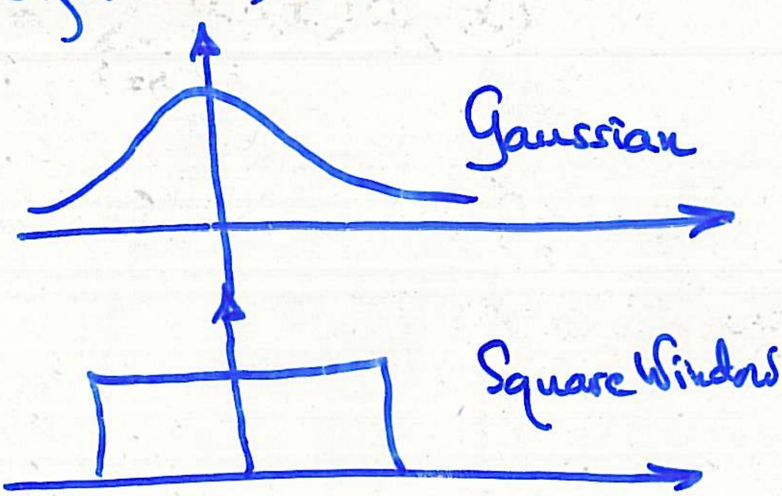
It can be the Least Squares fit

Distance $\left[f^{n}, f=\sum a_i \phi_i\right] =$

$$\int_{\Omega} \left(f^{n}(y) - \sum_{i=0}^{k} a_i(x_0)\, \phi_i(y)\right)^2 W(y-x_0)\, dy$$

where $W(\cdot)$ is a weight function that localizes the estimation of the coefficients $a_i(x_0)$.

Popular choices for $W(\cdot)$ are



If the data is samples $\{(x_i, f_i)\}$ then

$$\text{Distance}\left[\{f_i\}, \sum a_i \phi_i\right] =$$

$$= \sum_{i=1}^{k} \left(f_i - \sum_{i=0}^{k} a_i(x_0)\phi_i(x_i)\right)^2 W(x_i - x_0)$$

In both cases we mentioned the weighted least squares optimization process yields a nice formula for computing the local model coefficients $a_0(x_0), a_1(x_0) \ldots a_k(x_0)$ and the process of estimating the signal

$$\hat{f}(x) = \sum_{i=0}^{k} a_i(x)\phi_i(x)$$

is called the MOVING LEAST SQUARES ESTIMATION PROCESS.

So far we have a process of local modeling that yields from the data

$k+1$ functions

$$\{a_0(x), a_1(x), \ldots \; a_k(x)\}$$

which in turn provide the $\left\{\begin{matrix}\text{smoothed} \\ \text{denoised}\end{matrix}\right\}$ estimate of the signal $\hat{f}(x) = \sum_{i=0}^{k} a_i(x)\phi_i(x)$

But notice here that we have <u>not</u> <u>imposed any conditions</u> on the functions $a_i(x)$!

Notice also that in the definition of the neighborhoods via the weight functions $w(\cdot)$ we have not used any idea of adaptation to the <u>data</u>!

Global Constraints

Suppose we are given some structural information on the signal $f(x)$.
For example we might know that $f(x)$ is piecewise polynomial, i.e

$$f(x) = \sum_{i=1}^{k} \alpha_i^{(r)} x^i \quad \text{for } x \in [x_r, x_{r+1}]$$

In such a case we know a priori that if the $a_i(x)$ coefficients provide a good estimate for $f(x)$ then they'll be piecewise constant if $\phi_i(x)$ are chosen to be $[1, x, x^2, \dots x^k]$.

Hence the question naturally arises

WHY NOT IMPOSE SOME GLOBAL CONSTRAINTS ON $a_i(x)$'s to favor them to be piecewise constant!

Hence one can propose to supplement the MOVING LEAST SQUARES local fitting process with constraints on the varitions of parameters as we

MOVE!

Therefore we shall not only require the parameter vector $\theta = [a_0\, a_1 \ldots a_k]$ to provide best weighted local fit to the data but to also be consistent with local fits elsewhere!

This approach, imposing constraints of good local fitting to the data, and global smoothness on $k+1$-parameter functions $a_0(x) \ldots a_k(x)$

leads to the idea of OVERPARAMETERIZED variational methods.

We are now searching for $k+1$ functions $a_0(\cdot), \ldots a_k(\cdot)$ so that

$$\Psi_1[a_0 \ldots a_k] = \int_\Omega \int_\Omega \Big(f^n(y) - \sum_{i=0}^{k} a_i(x)\phi_i(y) \Big)^2 \underbrace{dy\,dx}_{w(y-x)}$$

$$+ \lambda \int_\Omega \sum_{i=1}^{k} \Big(\frac{d}{dx} a_i(x) \Big)^2 dx$$

NONLOCAL functional!

or

$$\Psi_2(a_0 \ldots a_k) = \int_\Omega \Big(f^n(x) - \sum a_i(x)\phi_i(x) \Big)^2 dx$$

$$+ \lambda \int \sum \Big| \frac{d}{dx} a_i(x) \Big| dx$$

and there are some other possibilities too like:

$$\Psi_3(a_0 \ldots a_k) = \int_\Lambda \left(f^u(x) - \sum_{i=1}^{k} a_i(x)\phi_i(x)\right)^{2p} dx$$

$$+ \lambda \int \sqrt{\sum \left|\frac{d}{dx} a_i(x)\right|^2}\, dx$$

One may regard overparametrization as a process of enlarging the space of possibilities where we look to get an interpretation for the noisy data we have, or alternatively as fitting better some prior data we have on the signals we are trying to estimate!

There are many extensions to be done:

- from uni+bivariate functions to curves and surfaces
- from local least squares fits to other distance functions
- from fixed weight functions to adaptive neighborhoods

CONCLUSIONS

There are many approaches to smoothing, denoising or data analysis, and to scattered data interpolation but many of them are related and their joint power should be better known and exploited...

On Globally Optimal Local Modeling: From Moving Least Squares To Over-Parametrization

Shachar Shem-Tov, Guy Rosman,
Gilad Adiv, Ron Kimmel and Alfred M. Bruckstein *

Abstract - This paper discusses a variational methodology, which involves locally modeling of data from noisy samples, combined with global model parameter regularization. We show that this methodology encompasses many previously proposed algorithms, from the celebrated moving least squares methods to the globally optimal over-parametrization methods recently published for smoothing and optic flow estimation. However, the unified look at the range of problems and methods previously considered also suggests a wealth of novel global functionals and local modeling possibilities. Specifically, we show that a new non-local variational functional provided by this methodology greatly improves robustness and accuracy in local model recovery compared to previous methods. The proposed methodology may be viewed as a basis for a general framework for addressing a variety of common problem domains in signal and image processing and analysis, such as denoising, adaptive smoothing, reconstruction and segmentation.

Shachar Shem-Tov, Guy Rosman, Ron Kimmel, Alfred M. Bruckstein
Technion - Israel Institute of Technology, 2000 Haifa, Israel,
e-mail: shemtov/rosman/ron/freddy@cs.technion.ac.il

Gilad Adiv
Rafael, Israel, e-mail: gilad3a@gmail.com

* This research was supported by the Israel Science foundation (ISF) grant no. 1551/09.

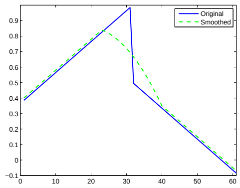


Figure 1: Solid line - y_s a piecewise linear signal with one discontinuity. Dashed line - a smooth version of y_s with which we initialized the model parameters

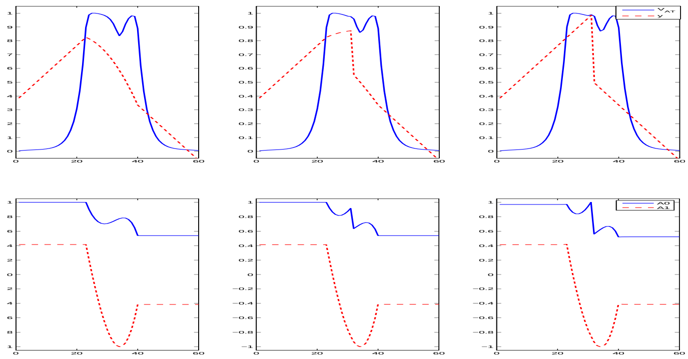


Figure 2: From left to right, snapshots at various times, of the MOP functional with relative weight of E_s $\alpha = 0.05$. The top image displays the reconstructed signal and the v_{AT} indicator function. The bottom image displays the reconstructed parameters.

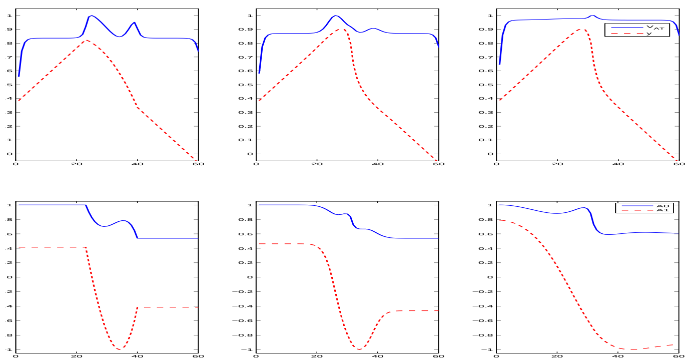


Figure 3: From left to right, snapshots at various times, of the MOP functional with relative weight of E_s $\alpha = 10$. The top image displays the reconstructed signal and the v_{AT} indicator function. The bottom image displays the reconstructed parameters.

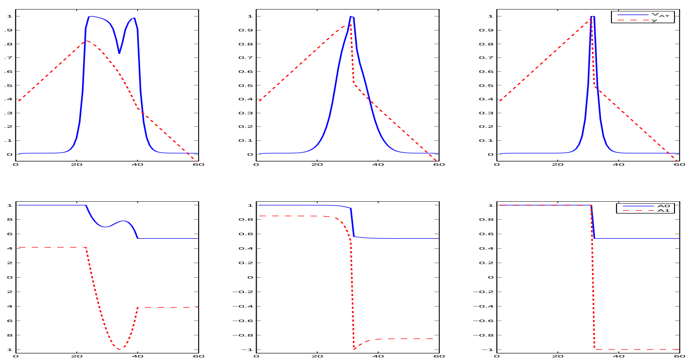
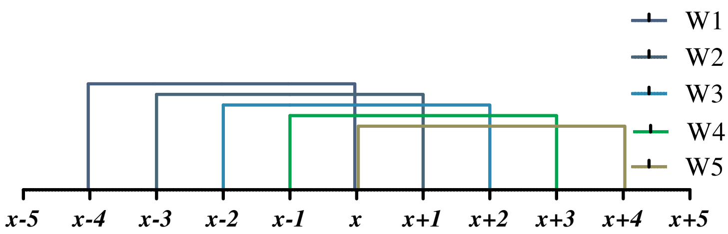
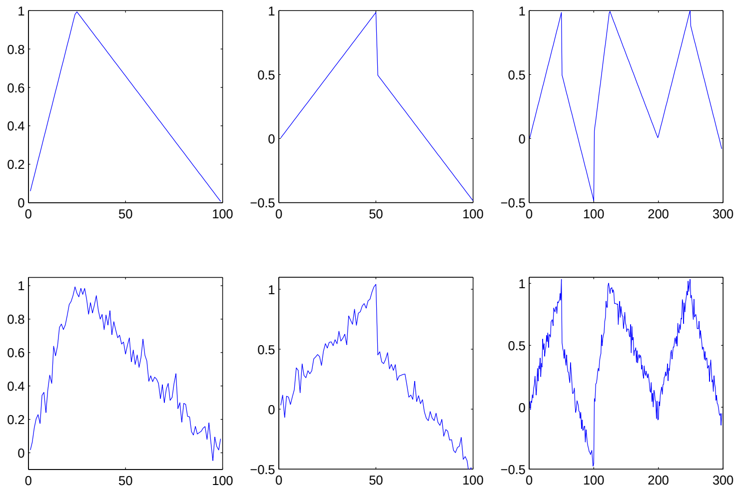


Figure 4: From left to right, snapshots at various times, of the NLOP functional. The top image displays the reconstructed signal and the v_{AT} indicator function. The bottom image displays the reconstructed parameters.



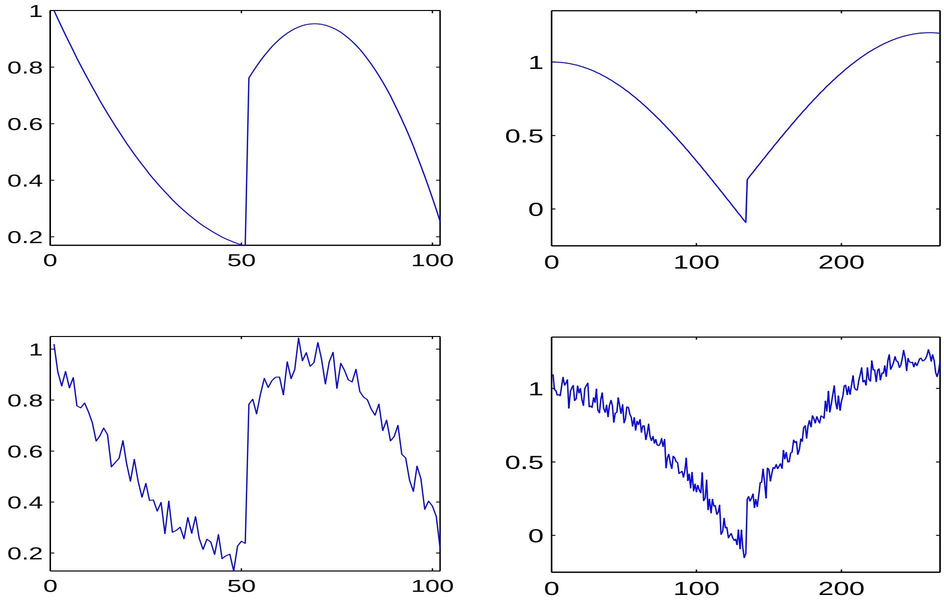


(a) One jump

(b) One discontinuity

(c) Many jumps & discontinuities

Figure 5: The piecewise Linear test signals and their STD 0.05 noisy counterparts.



(a) Polynomial

(b) Trig

Figure 6: The nonlinear test signals and their STD 0.05 noisy counterparts. Left: a polynomials signal of degree 2. Right: a signal of combined sine and cosine functions.

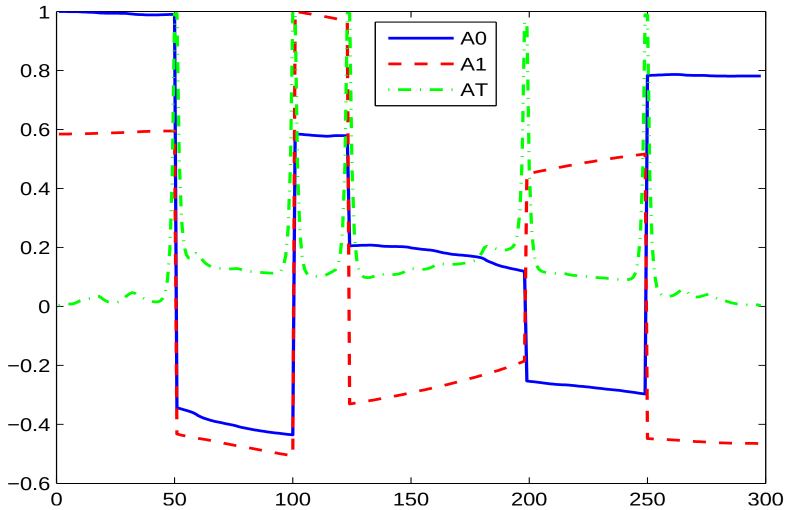
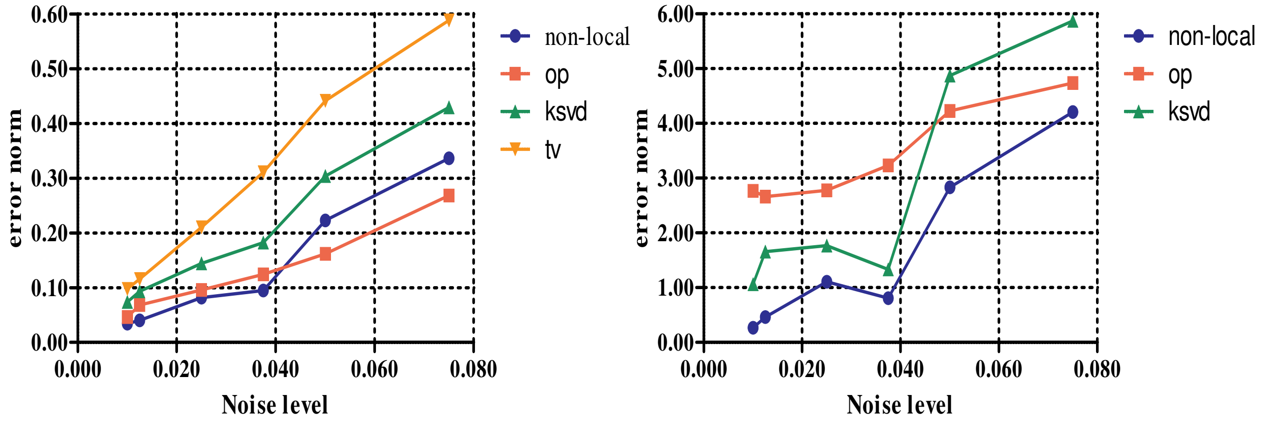


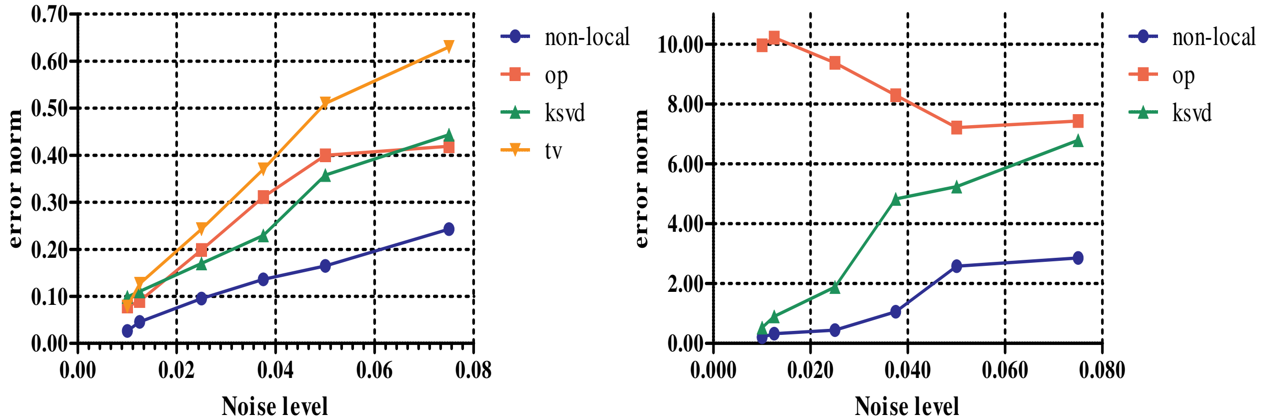
Figure 7: Example of parameters reconstruction. Note that the indicator function tends to 1 where the signal has parameters discontinuities and tends to 0 in almost any other location.



(a) One jump signal

(b) One jump parameters

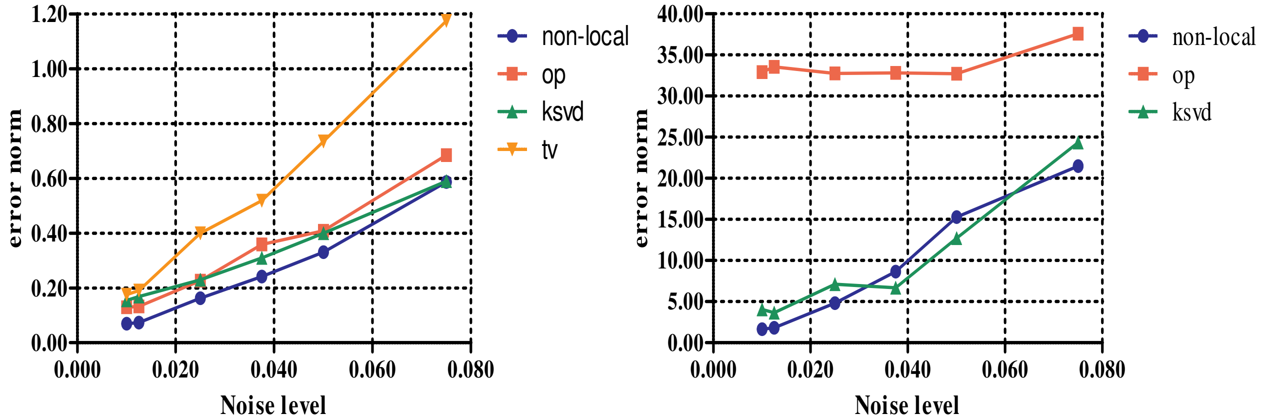
Figure 8: Signal noise removal and parameters reconstruction comparison for the ”One jump” signal.



(a) One discontinuity signal

(b) One discontinuity parameters

Figure 9: Signal noise removal and parameters reconstruction comparison for the ”One discontinuity” signal.



(a) Many jumps & discontinuities signal

(b) Many jumps & discontinuities parameters

Figure 10: Signal noise removal and parameters reconstruction comparison for the ”Many jumps & discontinuities” signal.

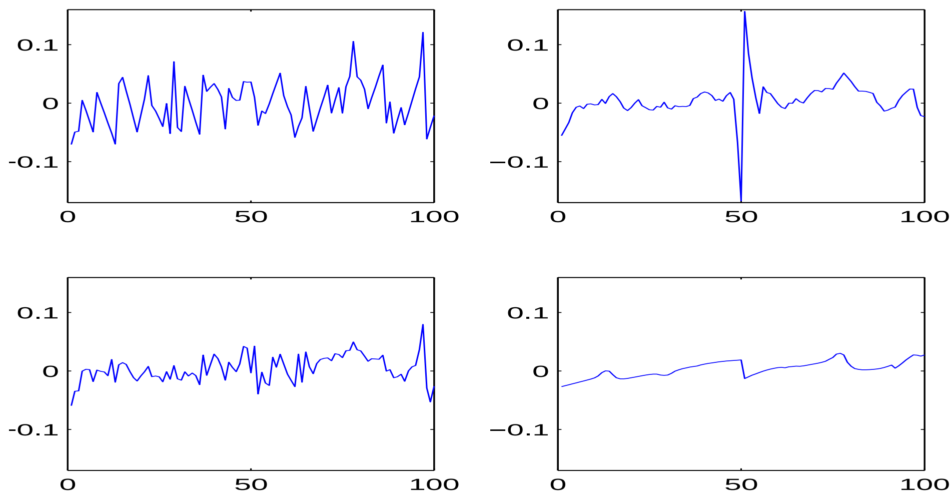


Figure 11: Comparison of noise removal on one discontinuity signal with noise std = 0.0375. Depicted are the residual noise images. Top left: TV. Top right: OP. Bottom left: K-SVD. Bottom right: NLOP.

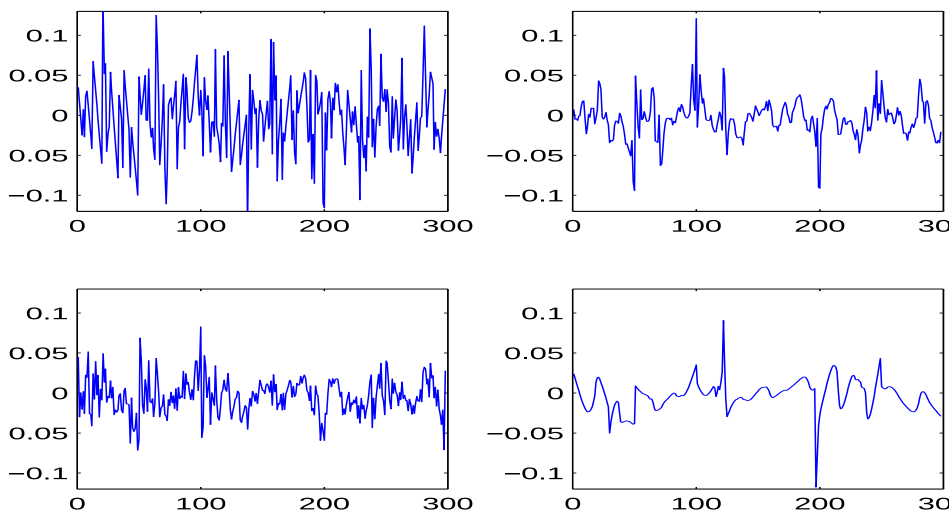
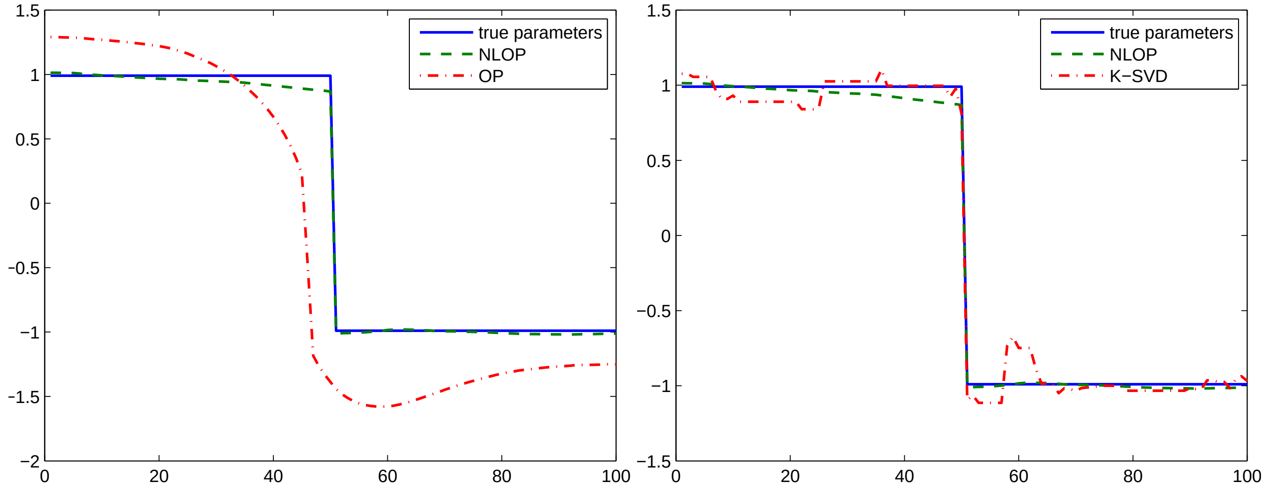
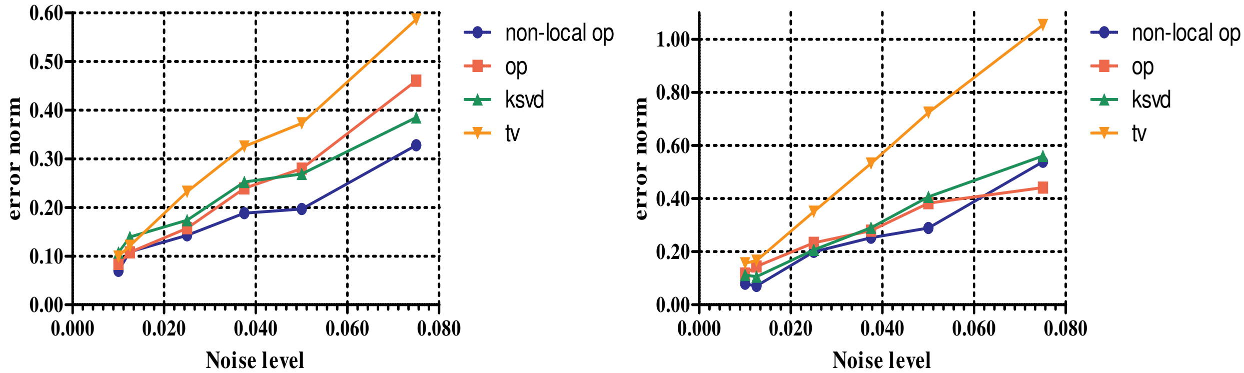


Figure 12: Comparison of noise removal, on the "many jumps & discontinuities" signal with noise std = 0.05. Depicted are the residual noise images. Top left: TV. Top right: OP. Bottom left: K-SVD. Bottom right: NLOP.



(a) NLOP vs OP (b) NLOP vs K-SVD

Figure 13: This figure compares the reconstructed second parameters on the various algorithms, when denoising the "One discontinuity" signal. Left image: Comparison of parameter reconstruction between NLOP functional and OP functional on one discontinuity signal. Note how far the OP reconstruction is from a piecewise solution, while generating an excellent denoising result (seen in the relevant graph). Right image: Comparison of parameter reconstruction between NLOP functional and K-SVD algorithm on one discontinuity signal. Note the apparent lack of global constraint on the parameters in the K-SVD reconstruction.



(a) polynomial signal (b) trig signal

Figure 14: Signal noise removal comparison of the nonlinear signals.

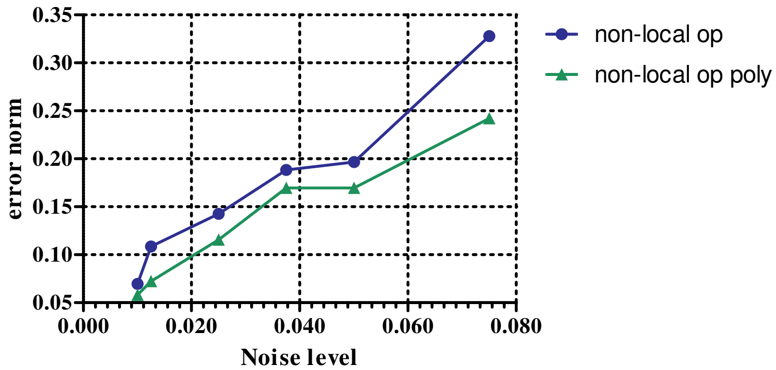


Figure 15: Comparison of noise removal performance on the polynomial signal. In this figure we compare performance of the NLOP functional with linear basis functions (marked by non-local OP), and NLOP functional with polynomial basis functions (marked by non-local OP poly).

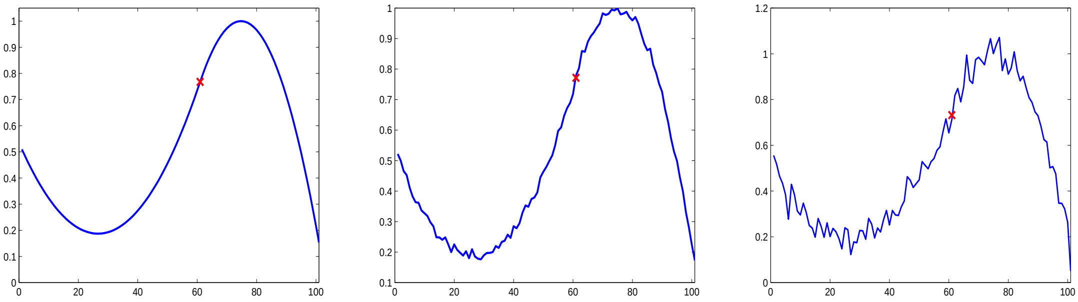


Figure 16: C_1 continuous 2nd degree polynomial. Point of the second derivative discontinuity is marked by a red cross. From left to right: clean signal, 0.01 STD noise, 0.0375 STD noise.

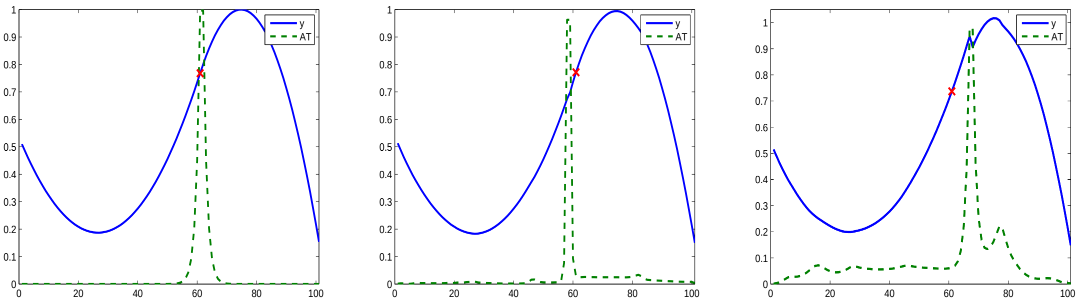


Figure 17: C_1 continuous 2nd degree polynomial reconstruction. Displayed are the reconstructed signals and the v_{AT} indicator functions. (point of the second derivative discontinuity is marked by a red cross). From left to right clean signal, 0.01 STD noise, 0.0375 STD noise.

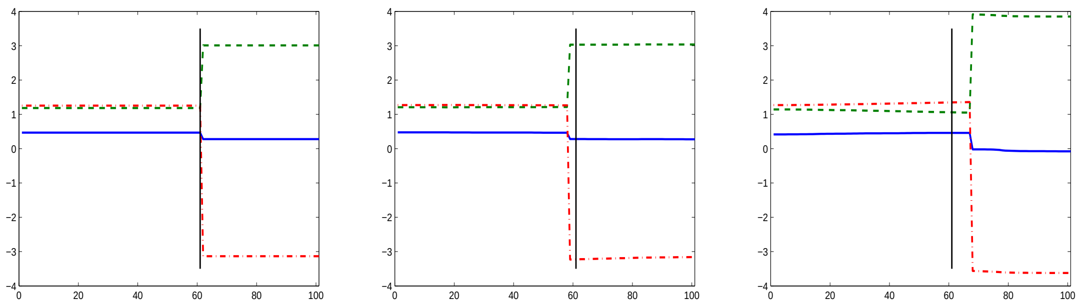
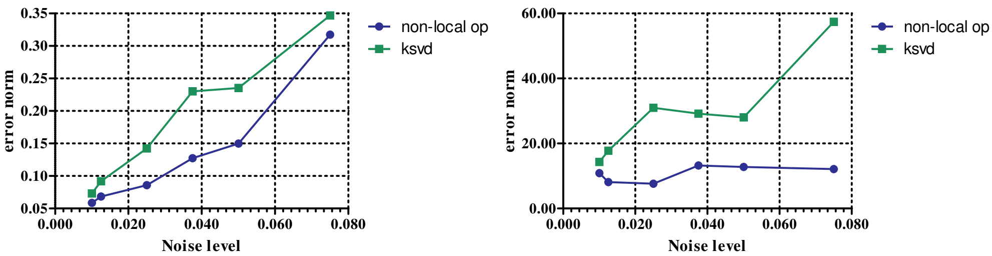
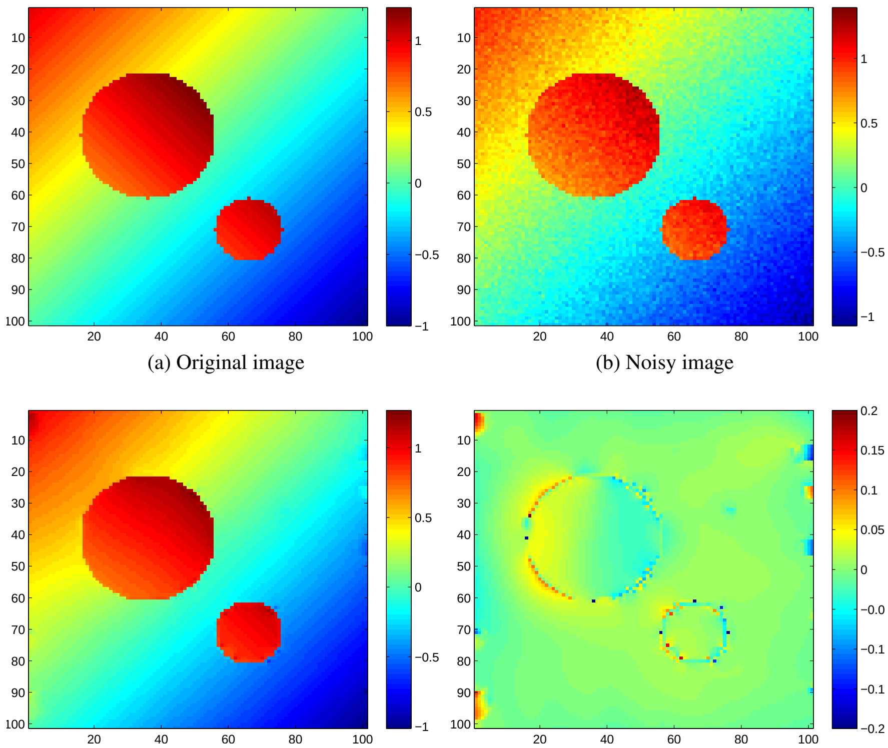


Figure 18: C_1 continuous 2nd degree polynomial reconstruction. Displayed are the reconstructed parameters (point of the second derivative discontinuity is marked vertical black line). From left to right clean signal, 0.01 STD noise, 0.0375 STD noise.



(a) Signal (b) Parameters

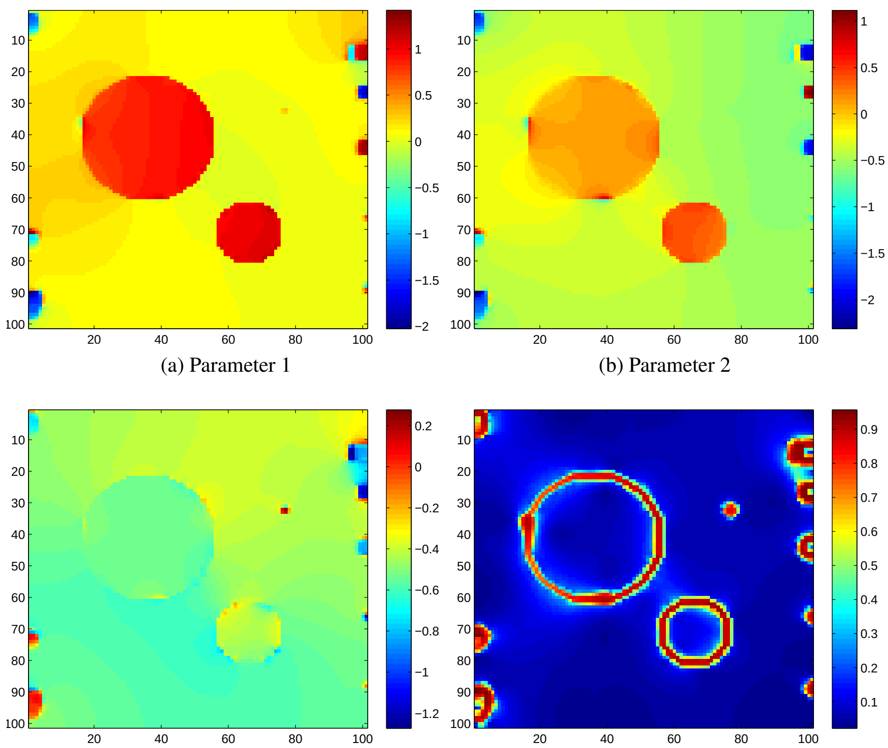
Figure 19: Signal noise removal and parameters reconstruction comparison for the C_1 continuous 2nd degree polynomial.



(a) Original image (b) Noisy image

(c) Denoised image (d) Residual noise

Figure 20: A 2D noise removal example of the 2D NLOP functional.



(a) Parameter 1 (b) Parameter 2

(c) Parameter 3 (d) v_{AT}

Figure 21: Example of the first two reconstructed parameters from the noisy image.