Savitzky-Golay Smoothing and Differentiation Filter Compared to Typical FIR Smoothing Filter (Nov 2009)

Samuel Harkness

***Abstract*— This document describes the design and implementation of a Savitzky-Golay smoothing filter in MATLAB.**

***Index Terms*— Savitzky-Golay, Smoothing, Filtering, Least Squares, MATLAB**

I. Introduction

THE Savitzky-Golay (SG) filter, also known as a polynomial smoothing or least-squares smoothing filter, is a low-pass filter well adapted to smoothing noisy data. [1] They are generalizations of the FIR average filter that can better preserve the high-frequency content of the desired signal, at the expense of not removing as much noise as the average.

II. Research

A. *Determining Filter Coefficients*

In order to achieve a high degree of noise reduction, the length L of a FIR average filter may be required to be so large the filter's passband $\omega_c = {}^{\pi}/_{L}$ becomes smaller than the signal bandwidth. [2] Past a certain size, the filter begins to smooth out the desired signal to an unacceptable degree.

Whereas a FIR averager fits a set of data points to a constant (a 0^{th} degree polynomial), a Savitzky-Golay filter fits a set of data points to an m^{th} degree polynomial (where m is generally >= 2). Therefore, the first step in designing a SG filter is determining the vector of coefficients $\vec{c}$ of the m^{th} degree polynomial fit function $\hat{x}_m[n]$.

$$\hat{x}_0[n] = c_0$$
$$\hat{x}_1[n] = c_0 + c_1 n$$
$$\hat{x}_2[n] = c_0 + c_1 n + c_2 n^2$$
$$\hat{x}_m[n] = c_0 + c_1 n + c_2 n^2 + \cdots + c_m n^m$$

Table 1: Example smoothing polynomial functions of degree m=0,1,2,m

The coefficients must be determined optimally such that the total mean-square error is minimized. This is accomplished by a least-squares fit. [3] The calculation of the mean-square error for a window of length L is:

$$\daleth = \sum_{n=0}^{L} e^2[n]$$

where the fitting error is defined as the difference between the input data and the fitted polynomial:

$$e[n] = x[n] - \hat{x}[n].$$

In matrix notation, $\hat{x}[n]$ can be written as

$$\hat{x}[n] = \overline{N}\vec{c}$$

where $\overline{N}$ is the matrix of the n^m vectors.

$$\overline{N} = [\vec{n}^0 \quad \vec{n}^1 \quad \vec{n}^2 \quad \cdots \quad \vec{n}^m]$$

$$\hat{x}[n] = c_0\vec{n}^0 + c_1\vec{n}^1 + c_2\vec{n}^2 + \cdots + c_m\vec{n}^m$$

It follows that

$$e[n] = x[n] - \hat{x}[n] = \vec{x} - \overline{N}\vec{c} = \vec{e}.$$

The mean-square error can then be written in matrix notation as

$$\begin{aligned} \daleth = \vec{e}^T\vec{e} &= \left(\vec{x} - \overline{N}\vec{c}\right)^T \cdot \left(\vec{x} - \overline{N}\vec{c}\right) \\ &= \vec{x}^T\vec{x} - 2\vec{c}^T\overline{N}^T\vec{x} + \vec{c}^T\overline{N}^T\overline{N}\vec{c} \end{aligned}$$

To minimize the expression with respect to $\vec{c}$, the gradient of the mean-square error is set to zero and solved for $\vec{c}$.

$$\begin{aligned} \frac{d\daleth}{d\vec{c}} &= -2\overline{N}^T\vec{x} + 2\overline{N}^T\overline{N}\vec{c} = 0 \\ &\xrightarrow{yields} \overline{N}^T\vec{x} = \overline{N}^T\overline{N}\vec{c} \end{aligned}$$

with optimal solution

$$\vec{c} = (\overline{N}^T\overline{N})^{-1}\overline{N}^T\vec{x}$$

Having solved for the polynomial coefficients, we insert this solution back into $\hat{x}[n]$

$$\hat{x}[n] = \overline{N}\vec{c} = \overline{N}(\overline{N}^T\overline{N})^{-1}\overline{N}^T\vec{x} = \overline{B}\vec{x}$$

where $\overline{B}$ is the matrix of FIR filter coefficients.

$$\overline{B} = \overline{N}(\overline{N}^T\overline{N})^{-1}\overline{N}^T$$

B. *Examination of FIR Filter Characteristics*

$\overline{B}$ is symmetric, meaning its rows are the same as its columns. Thus, it can be written in either column-wise or row-wise form.

$$\overline{B} = \begin{bmatrix}\vec{b}_0 & \vec{b}_1 & \vec{b}_2 & \cdots & \vec{b}_L\end{bmatrix} = \begin{bmatrix}\vec{b}_0 \\ \vec{b}_1 \\ \vec{b}_2 \\ \vdots \\ \vec{b}_L\end{bmatrix} = \overline{B}^T$$

Of all the columns of $\overline{B}$, the center filter is the most important because it is the steady-state Savitzky-Golay filter. The filters following the center are startup transient filters, while the filters preceding the center are ending transient filters. Because of the importance of the center filter, a requirement for SG filters is to be of an odd length.

Also, it is important to note it is a general property of Savitzky-Golay filters that the steady-state filter is the same for successive polynomial orders (i.e. $d = 0,1$ $d = 2,3$ $d = 4,5$). In other words, there is no odd ordered steady-state SG filter. However, the transient SG filters are different.

Finally, we construct the output of the filter, $y[n]$, as the convolution of the FIR filter, $h_i[n]$, convolved with the noisy input $x[n]$.

$$\begin{gathered} y_i[n] = h_i[n] * x[n] \\ = b_i[0] \cdot x[n] + b_i[1] \cdot x[n-1] + \cdots + b_i[L] \cdot x[n-L] \end{gathered}$$

A Savitzky-Golay filter is lowpass and is normalized to unity gain at DC, because its coefficients add up to one. This can be seen in the example below for a SG filter of degree $m = 2$ and length $L = 5$.

$$\vec{b}_2 = {}^{1}\!/_{35}\begin{bmatrix}-3 & 12 & 17 & 12 & -3\end{bmatrix}$$

The Noise Reduction Ratio (*NRR*) of a Savitzky-Golay filter is the sum of the squared filter coefficients. [2] It can be proven that the NRR of any steady-state SG filter is equal to the center value of its impulse response. For example, a 2nd order polynomial SG filter of length $L = 5$ has an NRR of:

$$NRR = \vec{b}_{L/2}^T \vec{b}_{L/2} = \sum_{i=0}^{L} \vec{b}_{L/2}^2 = \vec{b}_{L/2}\left[\frac{L}{2}\right] = .49$$

By comparison, a length 5 FIR average,

$$y[n] = \frac{1}{5}(x[n] + x[n-1] + x[n-2] + x[n-3] - x[n-4])$$

has $NRR = \frac{1}{L} = 1/5$. Thus, the length 5 Savitzky-Golay filter performs 2.43 times worse in reducing noise than the FIR averager. However, the SG filter has other advantages.

The larger the degree *m* of the filter, the flatter the response $B(\omega)$ becomes. This effectively increases the cutoff frequency of the lowpass – letting through more noise, but at the same time preserving more of the higher frequencies in the desired signal.

Figure 1and Figure 2 show the magnitude response of the steady-state filters of varying lengths and polynomial orders.

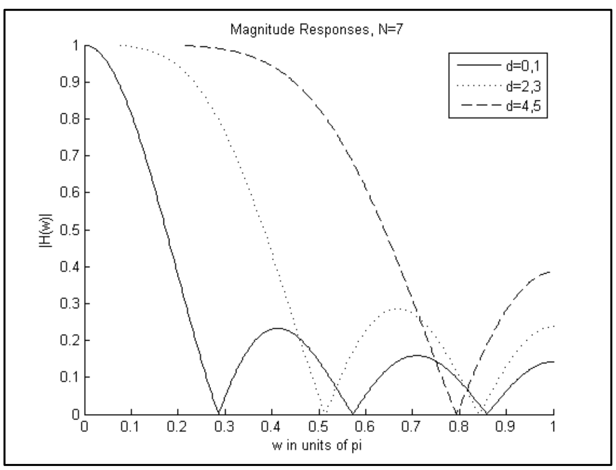

Figure 1: SG filters of length L = 7, and order m = 0, 2, 4

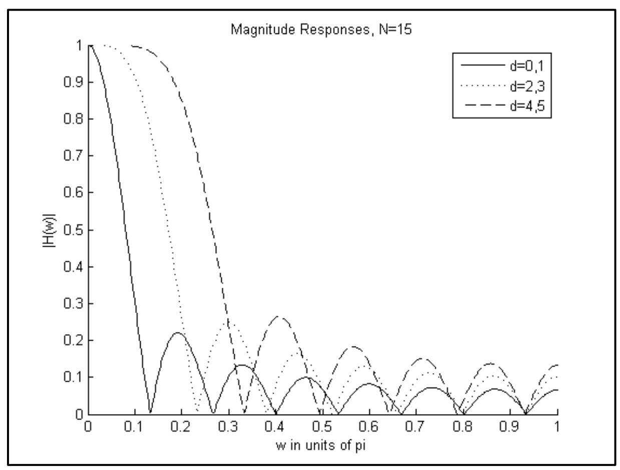

Figure 2: SG filters of length L = 15, and order m = 0, 2, 4

III. IMPLEMENTATION

A. *Filter Design*

For this implementation a length 7, 2nd order Savitzky-Golay filter was designed. This FIR filter would be compared to a 7-Point FIR average. The comparison was based upon preserved features of the filtered signals, such as relative maxima and minima. To demonstrate this comparison, filtering was performed on both an audio signal as well as an image.

The design of the 7-Point average is simple. Simply take the current value, and the previous six values, and average them out. This filter is denoted by the symbol b_{avg}.

$$b_{avg} = \frac{1}{7}[1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]$$

The MATLAB function sgolay(k,f) was used to build the Savitzky-Golay filter. [4] Input arguments were k=2 and f=7. This created an SG FIR filter of length 7 and polynomial order 2.

$$\overline{B} = \begin{bmatrix} .7619 & .3571 & .0714 & -.0952 & -.1429 & -.0714 & .1190 \\ .3571 & .2857 & .2143 & .1429 & .0714 & 0 & -.0714 \\ .0714 & .2143 & .2857 & .2857 & .2143 & .0714 & -.1429 \\ -.0952 & .1429 & .2857 & .3333 & .2857 & .1429 & -.0952 \\ -.1429 & .0714 & .2143 & .2857 & .2857 & .2143 & .0714 \\ -.0714 & 0 & .0714 & .1429 & .2143 & .2857 & .3571 \\ .1190 & -.0714 & -.1429 & -.0952 & .0714 & .3571 & .7619 \end{bmatrix}$$

However, the important filter is the steady-state. This filter is denoted by the symbol b_{sg}.

$$b_{sg} = [-.0952 \quad .1429 \quad .2857 \quad .3333 \quad .2857 \quad .1429 \quad -.0952]$$

Figure 3 shows the frequency response in both magnitude and phase of the two filters.

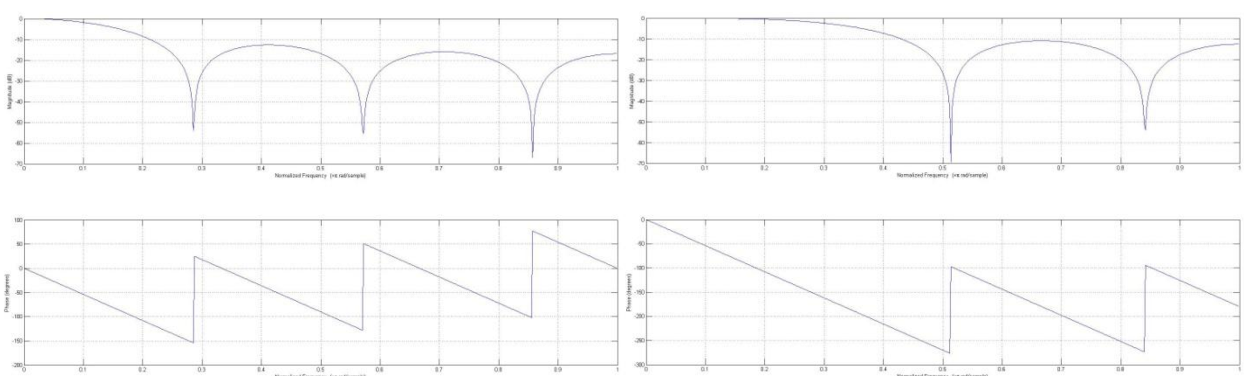

Figure 3: freqz results for b_{avg} and b_{sg}.

B. *Sound Processing*

A sound file playing the sentence “The big dog loved to chew on the old rag doll.” was used as the first point of comparison. The MATLAB function filter(b,a,x) was used for the filtering operation. Input argument a was set to 1 for both filtering operations. Figure 4 shows the original sound sample, as well as the filtered outputs. As can be seen, the Savitzky-Golay filter better preserves the local minima and maxima better than the 7-Point Averager. The sound samples can be heard using the attached MATLAB script.

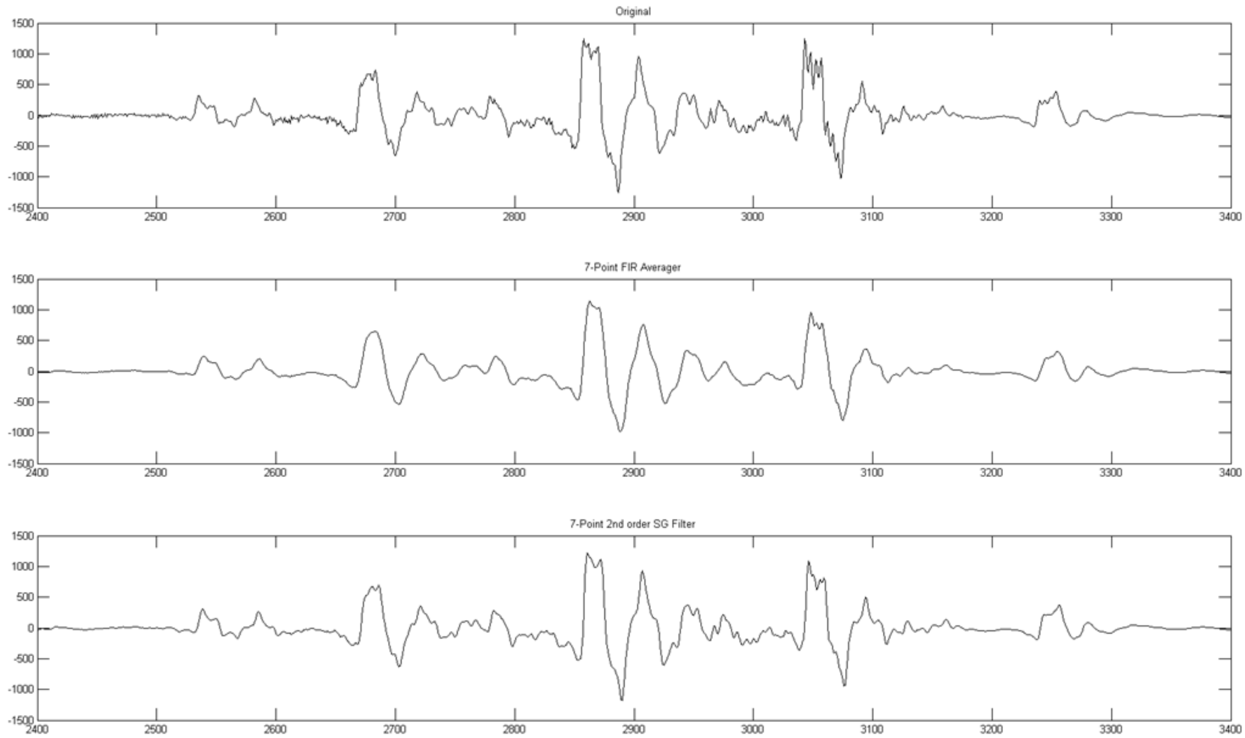

Figure 4: Original Sound Sample, and the Filtered Outputs

C. *Image Processing*

The image lenna from the DSPFirst toolkit was filtered. [5] The MATLAB function conv2(a,b) was used for the filtering operation. Figure 5 shows the original image, as well as the filtered outputs. Figure 6 shows a sample of the original image, as well as the filtered samples. As can be seen, the Savitzky-Golay filter better preserves the high frequency content over the 7-Point Averager.

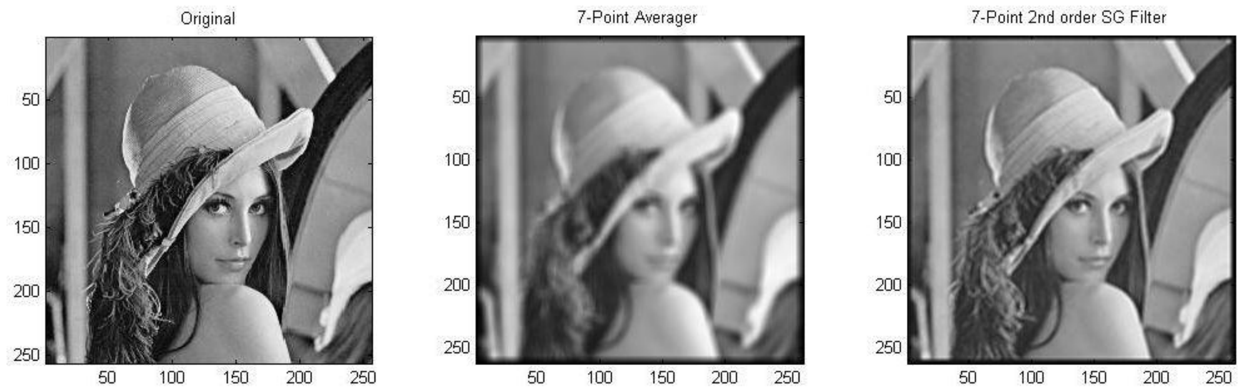

Figure 5: Original Image, and the Filtered Outputs

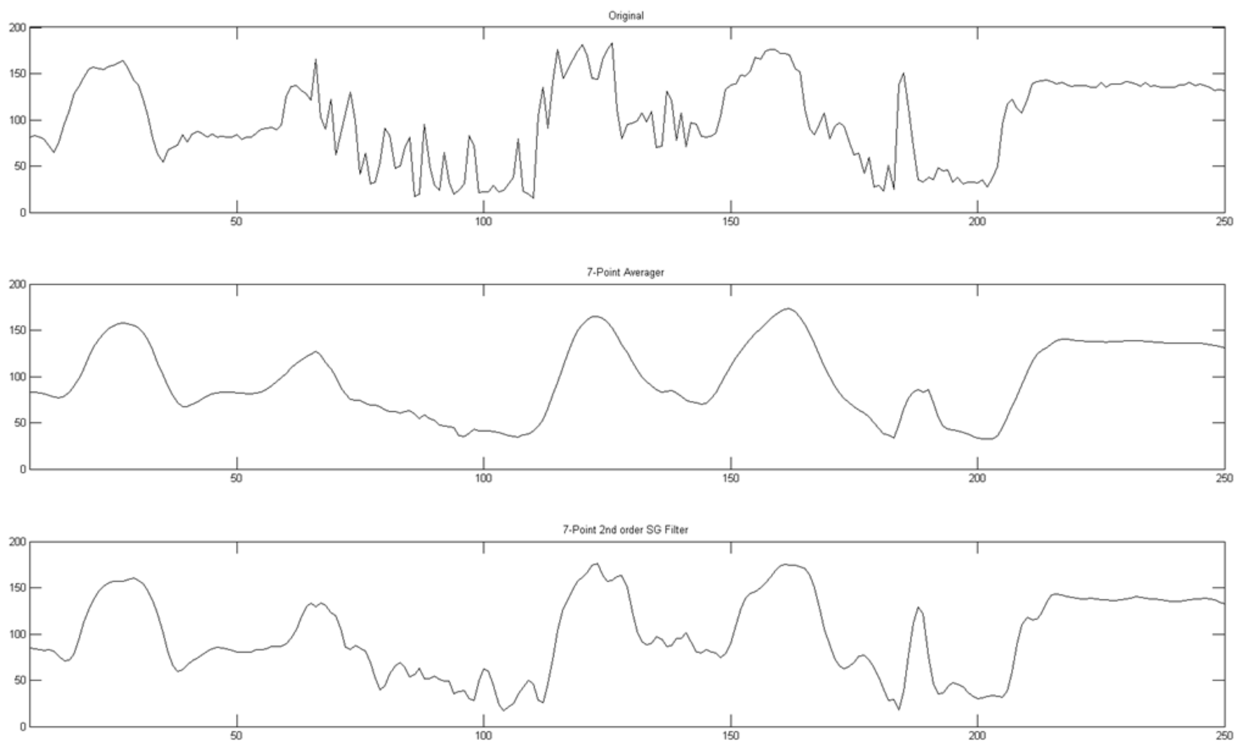

Figure 6: Original Image Sample, and the Filtered Outputs

ACKNOWLEDGMENT

The author thanks Robert C. Maher, PhD for the use of his copy of *Introduction to Signal Processing*. The author also thanks Ross K. Snider for providing the sound waveforms upon which this implementation was performed.

REFERENCES

[1] A. Savitzky and M.J.E. Golay, "Smoothing and Differentiation of Data by Simplified Least-Squares Procedures", *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964

[2] S.J. Orfanidis, "Savitzky-Golay Smoothing Filters," in *Introduction to Signal Processing*, A.V. Oppenheim, Ed. Englewood Cliffs: Prentice-Hall, 1996, pp. 434-459.

[3] J.L. Devore, "Estimating Model Parameters," in *Probability & Statistics for Engineering and Science*, Mason: Thomson, 2006, pp. 413-414.

[4] *MATLAB*. [CD-ROM]. Natick, Ma.: The MathWorks, 2008.

[5] J.H. McClellan, R.W. Schafer, and M.A. Yoder, *DSPFirst*. [CD-ROM]. Upper Saddle River, Nj.: Prentice Hall, 1998.

GLOSSARY

FIR	Finite Impulse Response
SG	Savitzky-Golay
m	The polynomial degree
L	The length of the filter
$x[n]$	The original noisy signal
$\vec{x}$	The vector notation of the original noisy signal
$\vec{x}[n]$	The fitted signal
$e[n]$	The error between the original signal and the fitted signal
$\daleth$	The mean-square error of the fitted signal
$\vec{n}^m$	The vector notation of the discrete time raised to the order m
$\overline{N}$	The matrix notation of the $\vec{n}$ vectors
$\overline{N}^T$	The matrix $\overline{N}$ has been mirrored across its diagonal
$\overline{B}$	The matrix of FIR filter coefficients
$\vec{b}_i$	The i^{th} vector from the matrix $\overline{B}$ of FIR filter coefficients
NRR	Noise Reduction Ratio
$y[n]$	The filtered signal